Trabajo Final de Ingeniería

Helper

Alumno: Natalia Mirta Gonzalez

Legajo: T1-5132

Sede: Castelar

Profesor: Jorge Scali

Índice

Trabajo Final de Ingeniería **1**

Índice **2**

1. Descripción del Negocio **6**

1.1 Descripción Básica del Negocio 6

1.2 Situación Actual del Negocio 6

1.3 ¿Qué hace único al negocio? 7

1.4 Factores Exitosos 9

1.5 Misión - Visión 10

1.5.1 Marco Temporal 10

1.5.4 Desafíos 13

1.5.5 Misión 13

1.5.6 Visión 14

1.6 Oportunidad 14

1.7 Propuesta de Valor 14

2. Análisis Estratégico **15**

2.1 Análisis de Contexto – Factores Externos 15

2.1.1 Descripción la Industria de Recursos Humanos 15

2.1.2 Factores Económicos 16

2.1.3 Factores Tecnológicos 20

2.1.4 Factor Legal 21

2.1.5 Factor Social 23

2.1.6 Factor Medio Ambiente 24

2.1.7 Oportunidades y Amenazas 24

2.2 Análisis Competitivo 26

2.2.1 Competidores Directos 26

2.2.2 Análisis Cadena de Valor de Servicios 27

2.2.3 Mi Cadena de Valor de Servicios 29

2.3 Comparación de Factores Críticos de Éxito 30

2.4 Fortalezas y Debilidades 33

3. F.O.D.A.....34

3.1 Cuadro F.O.D.A 34

3.2 Análisis F.O.D.A 35

4. Segmentación de Mercado 37

4.1 Criterios de Segmentación..... 37

4.1.1 Rentable..... 38

4.1.2 Accesible 38

4.1.3 Medible..... 39

4.1.4 Capacidad de Repuesta..... 40

4.2 Variables de Segmentación 41

4.2.1 Variables Demográficas..... 41

4.2.2 Variables Psicográficas..... 43

4.2.3 Variables Geográficas..... 44

4.2.4 Variables por Tasa de Uso..... 44

4.2.5 Variables por Beneficios 44

4.3 Mercado Meta 45

5. Estrategia de Negocio 46

5.1 Matriz Atractivo de la Industria / Fortaleza del Negocio 46

5.2 Programas de Generación de Acción 48

5.2.1 Programas de Administración..... 48

5.2.2 Programas de Servicios 49

5.2.3 Programa Estratégico de Marketing..... 50

5.2.4 Programa Estratégico Comercial..... 51

5.2.5 Programa Estratégico Tecnológico..... 52

5.2.6 Relación matriz FODA y programas estratégicos 53

5.3 Programas Específicos de Acción..... 54

5.3.1 Programa Generales..... 54

5.3.2 Programa Específicos de Administración..... 55

5.3.3 Programas Específicos de Comercial..... 61

5.3.4 Programa Específicos de Marketing 63

5.3.4 Programa Específicos de Servicios 64

5.2.5 Programa Específicos Tecnológico 66

5.4 Gantt..... 70

6. Plan de Marketing 71
6.1 Estrategia de Servicios 71
6.1.1 Descripción de los servicios 71
6.1.2 Características del servicio 72
6.1.3 Calidad del servicio 73
6.1.4 Líneas – mezclas 74
6.1.5 Ciclo de vida del Servicio 75
6.1.6 Estrategia de Marca 77
6.2 Estrategia de Precios 79
6.2.1 Análisis de precios 79
6.2.3 Estrategia de Precios 82
6.3 Estrategia de Promoción 83
6.3.1 Mix de Promoción 83
6.3.2 A.I.D. A 87
6.4 Estrategia de Distribución 89
6.4.1 Canales de Distribución 89
6.4.3 Funciones del Canal 89
6.4.3 Logística 90
7. Organización Requerida 91
7.1 Estructura Organizacional 91
7.2 Proceso de Toma de Decisiones 92
7.3 Proceso de tratamientos de conflictos 94
7.4 Análisis de Puesto 98
7.4.1 Gerente General 98
Gerente General 98
7.4.2 Encargada Administrativa 99
Encargada Administrativa 99
7.4.3 Asistente Administrativo Contable 100
Asistente Administrativo Contable 100
7.4.4 Jefe de Cobranzas 101
Jefe de Cobranza 101
7.4.5 Encargado Comercial 102
Encargado Comercial 102
7.4.6 Vendedor 103

Vendedor 103
7.4.7 Encargado de Servicios 104
Encargado de Servicios 104
7.4.8 Personal de Servicio 105
Personal de Servicios 105
7.4.9 Encargado de Tecnología - Desarrollador Full stack 105
Desarrollador Full stack 105
7.4.10 Encargado Analista de Datos 106
Encargado Analista de Datos 106
8. Plan Financiero **107**
8.1 Hipótesis 107
8.2 Modelo de ingresos 109
Gráfico Modelo de Ingresos 109
8.3 Modelo de Egresos 110
Gráfico Modelo de Egresos 110
8.4 Modelo de Inversión 111
8.4.1 Inversión año 0 112
8.4.2 Inversión año 1 112
8.4.3 Inversión año 2 113
8.4.4 Inversión año 3 113
8.5 Presupuesto Financiero 114
8.6 Matriz de Riesgos 115
8.7 Escenarios 117
8.8 Plan de Contingencia 119
8.8 Plan de Salida 121
9 Factibilidades 122
9.1 Técnica 122
9.2 Comercial 122
9.3 Financiera 123
9.4 Legal 124

1. Descripción del Negocio

1.1 Descripción Básica del Negocio

Somos un grupo de emprendedores que brindamos servicios de limpieza para el hogar. Hemos notado que existe un mercado con posibilidades de expandirse hacia aquellos consumidores que quieren un excelente servicio sin costos excesivos. Contamos con una amplia red de personas especializadas en limpieza doméstica, confiables, eficientes en zona oeste. La satisfacción de nuestros clientes nos abala, pero permítannos demostrarle con nuestro servicio.

Helper ofrece una pronta salida laboral que permite el manejo de horarios de manera flexible, y con el cobro de los servicios realizados por medios electrónicos. Deben contar con libreta sanitaria, certificado de antecedentes penales y monotributo. Se otorga capacitación, estabilidad y aseguradora de riesgo de trabajo.

1.2 Situación Actual del Negocio

Helper es una empresa joven dedicada a brindar servicio doméstico. Conformado por un equipo de trabajo competente que actúa hace poco más de dos años en Zona del Partido de Moreno, La Reja, Francisco Alvarez y General Rodríguez, Provincia de Buenos Aires.

Nos encontramos con una cartera de clientes muy exigentes. Dentro de los cuales contamos con algunos barrios cerrados y countries, por lo que nos vemos obligados a mantener un estándar de exigencia y un nivel de confianza óptima. Para cumplir con dichos objetivos, nuestro personal es capacitado y evaluado constantemente para poder ofrecer un servicio de calidad dentro del mercado.

Hasta el momento nuestra clientela se fue armando a través de las recomendaciones de los clientes. Nuestro objetivo es a través de la creación de un sitio web e-commerce el cual adoptara la modalidad B2C (empresa - consumidor) ya que se prestan servicios para clientes particulares. Mediante este nuevo canal de contacto que se crea un medio de comunicación para consultas, inquietudes, sugerencias, historial de consumos y contratación de servicio, a través del cual buscamos expandir nuestra cartera de clientes.

1.3 ¿Qué hace único al negocio?

Helper ofrece un servicio online de servicios de limpieza doméstica. Nuestro éxito se basa en las recomendaciones de nuestros clientes, por esta razón, somos muy exigentes y meticuloso con nuestro servicio y con el personal. Los servicios pueden ser programados desde la comodidad de su computadora o celular vía nuestro sitio web *Helper*, especificando servicio solicitado, horario y fecha. Simplemente cargando los datos en un formulario indicando domicilio, barrio e indicar porque medio electrónico va a abonar (tarjeta de crédito o transferencia bancaria). Garantizando de esta manera claridad en los servicios solicitados y evitando contratiempos.

En esta zona no hay agencias de personal doméstico, sin embargo, existen diferentes páginas que ofrecen empleos para tales tareas:

- Servicios domestico Lujan se encuentra a 20 km, pero ofrece servicios a alrededores http://www.sdomesticoslujan.com.ar/
- Zolvers tiene sus oficinas en CABA, pero ofrece servicios en zona oeste http://www.zolvers.com
- Portal de empleadas domestica http://www.portalempleadas.com.ar

Teniendo en cuenta lo remarcado anteriormente, Helper tiene las siguientes ventajas competitivas:

- *Exclusividad*: será la primera agencia en ofrecer un servicio doméstico en el distrito.

- *Buena reputación:* la empresa cuenta con un respetable historial, buena atención al público. Esta reputación confiable, respalda a *Helper* para poder lanzar este proyecto de expansión con mayor seguridad.

- *Precio competitivo*: se ofrecerá a los clientes la posibilidad de comprar productos a un precio más bajo que si se comprara en agencias que ofrecen servicios similares y facilidades de pagos con tarjetas.

- *Fidelidad de clientela*: según el historial de ventas, se ha verificado que un gran porcentaje de los clientes mantienen una fidelidad a la empresa ya que registran compras de los distintos servicios que ofrece *Helper*.

1.4 Factores Exitosos

Los puntos que destacan nuestros factores de éxito son:

- *Diferenciación*: Somos lo única agencia de la zona que brinda servicio de limpieza doméstica. Respondiendo a los diferentes perfiles de nuestros clientes y adaptándonos a sus requerimientos.
- *Precios claros*: contamos con precios competitivos. Brindando promociones por cantidades de horas, pack de servicios y beneficios para clientes frecuentes.
- *Tecnología*: aplicada para lograr una relación más ágil con nuestros clientes y a clientes potenciales a través de un sitio web e-commerce. Además, podrán concretar la adquisición de su servicio de manera on line, llevar su historial de consumos, forma de pago electrónica para ofrecer un acuerdo claro y sin sorpresas. Consultas a través de nuestros formularios de presupuestos.
- *Seriedad*: valoramos su tiempo, nuestro personal es muy puntual e idóneo en nuestro servicio. Contamos con personal capacitado y confiable.
- *Atención personalizada:* en nuestra oficina o telefónicamente, para nosotros es muy importante recibir de nuestros clientes sus experiencias y recomendaciones por los servicios brindados ayudándonos a detectar nuestras fortalezas y debilidades, nuestros puntos a mejorar.

1.5 Misión - Visión

1.5.1 Marco Temporal

Somos una empresa joven que no cuenta con tanta inversión y debido a la volatilidad de las políticas económicas, la falta de políticas monetarias e institucionales a través del tiempo, la planificación se torna un desafío en Argentina. Por este motivo, se entiende que la empresa debe tener una estructura flexible y una capacidad de reacción rápida y eficaz. En consecuencia, se toma como un tiempo prudencial tres años para poder establecer las metas y objetivos que se detallaran en las próximas secciones.

1.5.2 Alcance

1.5.2.1 Alcance del Mercado

El alcance del Mercado debería representar con exactitud a los clientes que utilizan el servicio de Helper:

- *Tipo de Industria seleccionada:* el tipo de industria de Helper es personal eventual de limpieza de nivel doméstico.
- *Aspectos demográficos*: nuestra cartera de clientes es muy variada desde casas de permanencia estable a quintas de fines de semana con grandes espacios. En la zona se cuenta con potenciales clientes en barrios cerrados y countries donde son consumidores de este tipo de servicios.
- *Composición de Consumidores:* hombres y mujeres en general mayores de 30 años y familias, en ambos casos suele ser de clase media o superior.
- *Canales de Distribución*: los canales actuales de distribución del servicio son a través telefónico o personalmente.

Alcances Existentes del Mercado	_ _	_	E	+	+ +
Mujeres					●
Hombres			●		
Familias clase media			●		
Familias clase alta					●
Propietarios inquilinos Countries					●
Barrios cerrados					●

1.5.2.2 Alcance del Servicio

Helper es una empresa de limpieza doméstica de manera eventual, por algunas horas, o permanente, programando una rutina semanal, que apunta a un mercado doméstico en expansión. Nuestro público en su mayoría son familias de clase media. Sin embargo, nos adaptamos a diferentes tipos de clientes para limpieza general e incorporamos para aquellos que necesitan un servicio de limpieza más profunda como limpieza de alfombra, ventanales, encerados de pisos, limpieza de tapizados y cortinas, etc. con personal de especializado.

En un principio nuestro contacto era a través de folletos, publicidad en diarios locales y nuestra popularidad llego de boca en boca y telefónicamente, hoy apuntamos a un contacto más dinámico a través de la creación de un sitio web donde el cliente pueda reservar su servicio, cotizar, dejar su opinión ò pedir presupuesto, permitiendo generar una comunicación más fluida por medio de redes sociales. Mediante las cuales, además de mails, hacemos llegar a nuestros clientes ofertas y promociones.

Alcances Existentes del Servicio	_ _	_	E	+	+ +
▼ Limpieza					
Domestica por Hora					●
Domestica Diaria				●	
Domestica Semanal			●		
Alfombras					●
Cortinas					●
Encerado de pisos					●
Ventanales					●
▼ Comunicación Pagina Web					
Pedido de cotizaciones				●	
Sugerencias		●			
Consultas				●	
Contratación de Servicio				●	

1.5.2.3 Alcance Geográfico

Alcances geográfico existente del Servicio	_ _	_	E	+	+ +
▼ Zona Oeste					
General Rodríguez			●		
Francisco Álvarez					●
La Reja			●		
Paso del Rey		●			
Merlo Centro			●		
Lujan					●
San Antonio de Padua				●	
Castelar					●
Ituzaingó				●	
▼ Zona Norte					
Pilar					●

1.5.3 Competencias Únicas

Competencias Únicas Existentes	_ _	_	E	+	+ +
Somos la única agencia local					●
Precios competitivos			●		
Reconocimiento de la marca				●	
Limpieza de calidad					●
Adquisición del servicio a través del Sitio web					●
Expansión Geográfica				●	
Nuevos Servicios de limpieza profunda					●

1.5.4 Desafíos

- **Fortalecer el reconocimiento de nuestra marca**, superar las dificultades ante nuestros competidores, enfrentando los problemas en contexto económico y político del país.
- **Expandir la zona donde brindamos nuestro servicio**, estamos confiados, por la satisfacción de nuestros clientes, que el servicio tendrá tendencia positiva en la ampliación de cartera de consumidores.
- **Mantener precios competitivos y rápida adaptación en una economía fluctuante**, a pesar de los vaivenes de la situación económica mantenemos precios atractivos sin afectar la calidad del servicio.
- **Propiciar buena atención**, es esencial mantener un buen ambiente de trabajo, calidez en la atención ya sea desde la compañía y para la satisfacción del cliente en su domicilio.

1.5.5 Misión

Brindar un servicio de limpieza de excelencia a precios competitivos para todos los hogares. Brindando satisfacción, calidad, seguridad, con personal capacitado y confiable. Nuestro compromiso con los clientes es el principal capital de la empresa.

1.5.6 Visión

Empresa líder en servicio de limpieza doméstica a medida de las necesidades del cliente, ofreciendo una experiencia de calidad a precios competitivos. A través de la tecnología logramos un canal de comunicación con nuestra cartera de consumidores.

1.6 Oportunidad

En los tiempos que vivimos donde los momentos libres escasean y donde el funcionamiento de las familias se ha visto alterados estructuralmente, ya no suele estar esa imagen de la mujer de la casa limpiando y cocinando por horas, sino que los roles se han modificado.

La economía y los nuevos roles que han tomado las mujeres en la sociedad hacen que estén más horas fuera de casa y el servicio doméstico ya no es algo de las clases sociales más acaudaladas. Ofrecemos personal capacitado y de confianza para que nuestros clientes se sientan seguros.

1.7 Propuesta de Valor

Helper ofrece un servicio de calidad, seguro y de excelencia:

- **Personal capacitado**: limpieza eficiente, meticulosa, puntual, con productos óptimos respetando el medio hambre.
- **Flexibilidad**: ya sea por horas en una situación eventual, o programando visitas semanales, quincenales, mensuales, etc. Trabajamos de lunes a sábados.
- **Seguridad**: Información y referencias del personal, evaluada por la experiencia de los clientes.

2. Análisis Estratégico

2.1 Análisis de Contexto – Factores Externos

2.1.1 Descripción la Industria de Recursos Humanos

El Instituto Nacional de Estadísticas y Censos (INDEC) actualizó los registros de cálculo del **PBI** (Producto Bruto Interno) para el segundo trimestre del año con un descenso del 19,1% como consecuencia de la cuarentena más rígida para contener a la pandemia del coronavirus, y desde un nivel muy bajo luego de años acumulados de recesión e inestabilidad, la economía argentina mostró en el segundo trimestre del año el mayor derrumbe en toda su historia.

Ante este panorama los empleadores prefieren contratar su personal a través de agencias para no contraer una responsabilidad directa.

Las agencias de recursos humanos son una buena opción para los empleadores ya que ofrece agilidad en el reclutamiento, selección y personal. Asesoramiento respecto del perfil que mejor se adecúa a las necesidades relevadas. Alta efectividad entre la calificación del empleado y la necesidad de la usuaria. Administración eficiente del personal. Seguridad Jurídica. El régimen legal que regula la actividad garantiza que el personal temporario se encuentre debidamente registrado en todos los casos. Capacidad de cubrir grandes volúmenes de personal.

Por otro lado, si de la selección del personal se ocupara un encargado de la empresa se desgastaría ya que si no cuenta con la suficiente experiencia y no le sería una tarea sencilla elegir a la persona correcta. El trabajo de seleccionar personal no es una tarea simple ya que en los jóvenes se registran altos niveles de ausentismos en las entrevistas y tienen dificultades para expresarse correctamente o van desinformados sobre la empresa donde podrían ingresar.

Según el FAETT (Federación Argentina de Empresas de Trabajo Temporario) en marzo 2019 señala que las personas más jóvenes se encuentran en el segmento de la modalidad temporal o eventual. Aproximadamente el 60% son menores de 30 años, suelen ser sus primeras experiencias en el mundo laboral y les permite horarios flexibles. A través de la creación de empleos surge que 4 de cada 10 personas que ingresan en esta modalidad pasan a ser planta permanente de sus empleadores. Siendo los de nivel profesional de mayor permanencia con respecto a los operativos en los que se registra mayor rotación.

El impacto de la cuarentena

Más de 50.000 empleadas domésticas perdieron su trabajo en la pandemia. Los datos oficiales muestran que se dieron que unas 20.000 trabajadoras registradas dejaron de hacer los aportes al sistema previsional. La cifra aumenta notablemente si se consideran los puestos no declarados.

Según los datos del Ministerio de Trabajo, el mayor registro de empleo de trabajadoras domésticas en blanco fue de 501.000 en octubre 2019. Luego cayó a 497.000 en febrero de este año -prepandemia- y a 471.000 en mayo último. Para el Indec, en tanto, a marzo había un total 1,7 millones (1.696.145) de puestos de trabajo en hogares, de los cuales estaban registrados solo 511.354 casos.

2.1.2 Factores Económicos

2.1.2.1 Inflación

La inflación en el 2019 cerro en los 53,8 puntos, empujada fuertemente por el aumento de los servicios por la quita de subsidios, combustibles, durante diciembre los rubros Alimentos y Bebidas, con un incremento de 3,1%; Esparcimiento y Mantenimiento para el hogar, 5,4%; Salud, 5,6%; y Comunicación, 9 %, entre otros. Servicios domésticos del 10% en dos tramos (junio y diciembre) y prepagas.

En diciembre de 2019 se cambian las metas inflacionarias con la llegada del gobierno de Alberto Fernández, en políticas económicas para incentivar el crecimiento, lo que dispara incertidumbre ya que el aumento de los costos afecta el crecimiento en las industrias.

La agencia de empleo eventual ManpowerGroup llevó a cabo la Encuesta de Expectativas de Empleo para el segundo trimestre de 2020 a una muestra representativa de 800 empleadores de la Argentina. Los empleadores argentinos informan planes de contratación laboral moderados para los próximos tres meses. Un 8% de los empleadores prevé aumentar su dotación, un 4% anticipa una merma y un 80% no anticipa cambios en su plantilla, lo que arroja una ENE (expectativa neta de empleo) de +4%. Al ajustar los datos para tener en cuenta la variación estacional, la expectativa de empleo se coloca en +3%. La generación de empleo mejora 2 puntos porcentuales en comparación con el

trimestre anterior y 3 puntos porcentuales en comparación con el mismo período de hace un año.

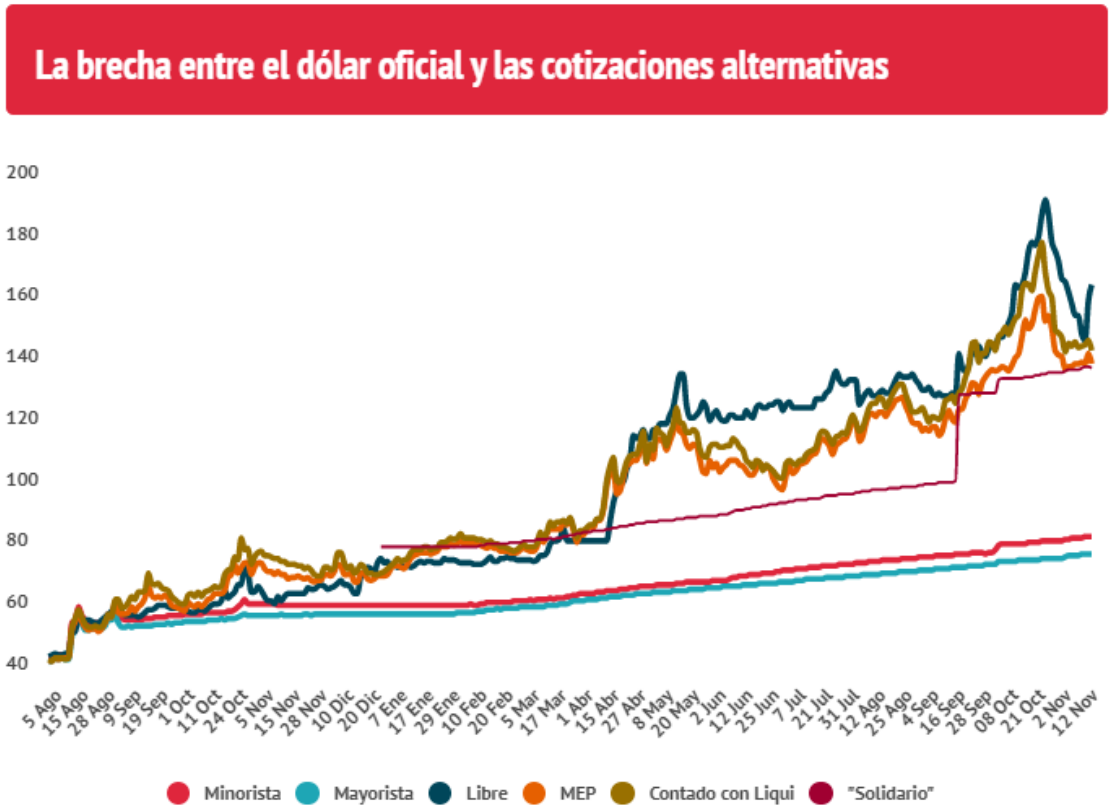
	Aumento	Disminución	Sin Cambios	No Sabe	Expectativa Neta de Empleo	Ajustada por Estacionalidad
	%	%	%	%	%	%
Abr-Jun 2020	**8**	**4**	**80**	**8**	**4**	**3**
Ene-Mar 2020	7	6	83	4	1	1
Oct-Dic 2019	8	6	82	4	2	2
Jul-Sep 2019	8	8	81	3	0	1
Abr-Jun 2019	9	8	79	4	1	0

2.1.2.2 Tipo de Cambio

La falta de medidas económicas convincentes influyó en las valuaciones de los activos en las últimas semanas. Las cotizaciones en dólares y acciones se desplomaron, volvió a subir el riesgo país y los dólares bursátiles alcanzaron precios récords como efecto de la búsqueda de cobertura de los inversores por lo cual el Banco Central tuvo que intervenir vendiendo más de u$s 50 millones.

En el último mes el dólar contado con liqui subió 29%, a un máximo de 166 pesos y la brecha cambiaria con el oficial mayorista supero el 100 por ciento (ahora en el 114%). El rango más alto en tres décadas. El 15 de septiembre esa brecha era del 70 por ciento.

El aumento en la cotización del dólar estadounidense provoca una devaluación sobre el peso argentino causando una perdida en el poder adquisitivo.


La brecha entre el dólar oficial y las cotizaciones alternativas
200
180
160
140
120
100
80
60
40
5 Ago
15 Ago
28 Ago
9 Sep
19 Sep
1 Oct
11 Oct
24 Oct
5 Nov
15 Nov
28 Nov
10 Dic
20 Dic
7 Ene
17 Ene
29 Ene
10 Feb
20 Feb
5 Mar
17 Mar
1 Abr
15 Abr
27 Abr
8 May
20 May
2 Jun
12 Jun
25 Jun
7 Jul
21 Jul
31 Jul
12 Ago
25 Ago
4 Sep
16 Sep
28 Sep
08 Oct
21 Oct
2 Nov
12 Nov
Minorista
Mayorista
Libre
MEP
Contado con Liqui
"Solidario"

2.1.2.3 Impuestos

Registrar un empleado o empleada de servicio doméstico tiene un costo diferente para el empleador de acuerdo con la cantidad de horas que trabaje. Además, los aportes y contribuciones aumentaron a partir de enero 2020, por lo que el costo será mayor.

Las contribuciones para el servicio doméstico tienen un nuevo valor nominal suba que va del 14% al 31% a partir del sueldo de febrero.

Personas que trabajen menos de 12 horas semanales:

- ✓ Aumento en las contribuciones y aportes (a cargo del empleador) del 14%.
- ✓ El aporte mensual de obra social quedará en $84,50
- ✓ La contribución quedará en $29,80.
- ✓ La ART seguirá en $196.
- ✓ **El costo total para el empleador será de $310,30.**

Personas que trabajan entre 12 y 16 horas semanales:

- ✓ Aumento en las contribuciones y aportes (a cargo del empleador) del 16%
- ✓ El aporte mensual de obra social quedará en $156,50.
- ✓ La contribución quedará en $59,60.
- ✓ La ART será de $314
- ✓ **El costo total para el empleador será de $530,10.**

Personas que trabajan durante 16 horas semanales o más:

- ✓ Aumento en las contribuciones y aportes (a cargo del empleador) del 31%
- ✓ El aporte mensual de la obra social quedará en $1041,20
- ✓ La contribución quedará en $87.
- ✓ La ART será de $484
- ✓ **El costo total para el empleador será de $1612,20.**

Factores Económicos

Actual Futuro	Muy poco atractivo	Poco atractivo	Neutro	Atractivo	Muy atractivo
Inflación					
Tasa de Interés					
Tipo de Cambio					
Impuestos					

2.1.3 Factores Tecnológicos

La era tecnológica de comunicación ha facilitado enormemente a la industria a la hora de contactar a las personas. Tanto sea para postularse a empleos, acceder a currículums de cientos de personas según los perfiles de búsquedas o publicar ofertas laborales. Ganando tiempo ya sea para las agencias como para los postulantes. Se puede tener en un corto lapso respuestas a una propuesta laboral y cubrir la vacante. Algunas hasta realizan entrevistas a través de Skype, zoom, Google meet, entre otras.

Por otro lado, las empresas se ven obligadas a adaptarse a los tiempos que corren, si no figuran en la red es como si no existieran. Pero no todo lo que brilla es oro, porque nos podemos encontrar con muchos currículums, debemos estar muy atentos a que sean recientes y actualizados y que sean postulantes activos.

Factores Tecnológicos

Actual Futuro	Muy Poco Atractivo	Poco Atractiv0	Neutro	Atractivo	Muy Atractivo
Internet					
Hardware					
Software					
Telecomunicaciones					

2.1.4 Factor Legal

2.1.4.1 Leyes

SERVICIO DOMESTICO Ley 26.844

ARTICULO 1° — Ámbito de aplicación. La presente ley regirá en todo el territorio de la Nación las relaciones laborales que se entablen con los empleados y empleadas por el trabajo que presten en las casas particulares o en el ámbito de la vida familiar y que no importe para el empleador lucro o beneficio económico directo, cualquiera fuere la cantidad de horas diarias o de jornadas semanales en que sean ocupados para tales labores.

ARTICULO 2° — Aplicabilidad. Se considerará trabajo en casas particulares a toda prestación de servicios o ejecución de tareas de limpieza

ARTICULO 3° — Exclusiones - Prohibiciones. No se considerará personal de casas particulares y en consecuencia quedarán excluidas del régimen especial.

ARTICULO 4° — Principios de interpretación y aplicación de la ley. Cuando una cuestión no pueda resolverse por aplicación de las normas que regulan el presente régimen, se decidirá conforme a los principios de la justicia social, a los generales del derecho del trabajo, la equidad y la buena fe.

ARTICULO 5° — Grupo familiar. Retribución. En caso de contratarse más de una persona de la misma familia para prestar servicios a las órdenes de un mismo empleador, la retribución deberá convenirse individualmente con cada uno de ellos.

ARTICULO 6° — Contrato de trabajo. Libertad de formas. Presunción. En la celebración del contrato de trabajo para el personal de casas particulares regirá la libertad de formas cualquiera sea su modalidad. El contrato se presumirá concertado por tiempo indeterminado.

ARTICULO 7° — Período de prueba. El contrato regulado por esta ley se entenderá celebrado a prueba durante los primeros treinta (30) días de su vigencia respecto del personal sin retiro; y durante los primeros quince (15) días de trabajo en tanto no supere los tres (3) meses para el personal con retiro. Cualquiera de las partes podrá extinguir la relación durante ese lapso sin expresión de causa y sin generarse derecho a indemnización con motivo de la extinción. El empleador no podrá contratar a una misma empleada/o más de una (1) vez utilizando el período de prueba.

ARTICULO 8° — Categorías profesionales. Las categorías profesionales y puestos de trabajo para el personal comprendido en el presente régimen serán fijadas inicialmente por la autoridad de aplicación hasta tanto sean establecidas por la Comisión Nacional de Trabajo en Casas Particulares o mediante convenio colectivo de trabajo.

Links de Interés:

- Ley completa :26.844 http://servicios.infoleg.gob.ar/infolegInternet/anexos/210000-214999/210489/norma.html
- AFIP Casas particulares: https://casasparticulares.afip.gob.ar/default.aspx
- Preguntas frecuentes sobre el empleo de servicio domestico en Argentina https://www.argentina.gob.ar/trabajo/domestico/preguntas

Actual Futuro	Muy Poco Atractivo	Poco Atractiv0	Neutro	Atractivo	Muy Atractivo
Leyes Laborales					

2.1.5 Factor Social

Según el último informe del INDEC, la pobreza afecta al 47 por ciento de la población, pero son cinco las provincias que superan esta tasa: Buenos Aires con el 53,9, Chaco con el 50,8%, Salta con 49,9%, Río Negro con el 48,4% y Neuquén con 47,5%. A la vez, las primeras cuatro superan la tasa de indigencia nacional.

En ese sentido, el Instituto de Pensamiento y Políticas Públicas presentó “El mapa argentino del Hambre y la Pobreza”, donde analizan los datos del segundo trimestre del 2020. Su coordinador Claudio Lozano, quien además es Director de Banco Nación y dirigente de Unidad Popular, explicó que “el INDEC presenta datos semestrales, nosotros los seguimos trimestralmente”, y que con la pandemia y la cuarentena hubo un salto en términos de pobreza en el segundo trimestre.

“Tenemos 21 millones de personas pobres en la Argentina” explicó el economista. En los primeros tres meses de 2020 la pobreza estaba en el 34 por ciento, y en solo un trimestre, 5.600.000 personas cayeron bajo la línea de pobreza y otras 1.700.000 personas bajo la línea de indigencia.

Lo que refleja perdida del empleo y un descenso de la generación de empleos ya sea de manera formal e informal.

Actual Futuro	Muy Poco Atractivo	Poco Atractiv0	Neutro	Atractivo	Muy Atractivo
Factor Social					

2.1.6 Factor Medio Ambiente

En la industria de recursos humanos el factor medio ambiental que más afecta al presentismo es el climatológico sobre todo en épocas invernales, ya que con las bajas temperaturas afloran las enfermedades virales.

Por otro lado, si nos referimos a medioambientales humanos debemos tener en cuenta que se refiere a tener un personal capacitado, motivados y que cuenten con incentivos, que tengan ideas claras la organización jerárquica de la empresa y sus relaciones de autoridad.

Actual Futuro	Muy Poco Atractivo	Poco Atractiv0	Neutro	Atractivo	Muy Atractivo
Factor Medio Ambiental					

2.1.7 Oportunidades y Amenazas

El nivel de inflación, la suba en las tarifas de los servicios, el aislamiento debido a la pandemia, el aumento en el precio del combustible son factores que afectan el nivel de consumo de las personas y que hace que descarten primeramente aquellas cosas que no son esenciales. Esta situación afecta a la industria.

Oportunidades

- La negociación de paritarias y que en el nuevo contexto de las familias más de un miembro trabaja permite que dichos sectores hagan esfuerzos para seguir manteniendo su nivel y calidad de vida.
- El uso de nuevas tecnologías y canales de comunicación generan un incremento en la cantidad de clientes.
- Con la crisis económica las familias prefieren contratar personal eventualmente y generarse un compromiso directo al contratarlo.

Amenazas:

- Los vaivenes económicos, el costo de los servicios, los niveles inflacionarios elevados, caída en el consumo. En lo que va del año el tipo de cambio oficial del peso retrocedió 22% con respecto al dólar, una caída que se aceleró a principios de octubre tras la introducción de controles de capital más estrictos. Todos estos factores retrotraen la generación de nuevos puestos.
- El aumento de los salarios y la presión tributaria pueden afectar a la generación de horas extras de trabajo.
- Industria del juicio laboral, el gran incremento en las demandas a las empresas y ART a afecta a la generación de nuevos empleos. Se estima que la “industria del juicio” genero más de 68.898 juicios en 2019, lo cual implica en costos de seguro en la alícuota de la ART. Frente a un juicio el empleado al ser la parte débil suele ser beneficiada.

Actual Futuro	Muy Poco Atractivo	Poco Atractiv0	Neutro	Atractivo	Muy Atractivo
Análisis de la Industria					

2.2 Análisis Competitivo

2.2.1 Competidores Directos

- Zolvers empleadas domésticas y servicios para el hogar ubicación de oficinas en Barrio Belgrano - CABA https://zolvers.com/
- Servicio Doméstico Lujan http://www.sdomesticoslujan.com.ar/ que no solo ofrece servicio doméstico, sino que además cocinero, niñera, enfermero, caseros, etc.

2.2.2 Análisis Cadena de Valor de Servicios

2.2.2.1 Analizar cadena de valor – Zolver

Dirección General y Recursos Humanos						**Margen del Servicio**
Organización Interna y de Tecnología						
Infraestructura y ambiente						
Abastecimiento						
Marketing y ventas Facebook Twitter	**Personal de Contacto** Todos cuentan con experiencia compro-bada, refe-rencias y son verificados por nuestro equipo.	**Soporte Físico y habilidades** Sitio web	**Presentación** Fuerte presen-cia en CABA servicio doméstico y mantenimiento del hogar.	**Clientes** Cuentan con presen-cia en Argentina, Chile, México y Colombia	**Otros Clientes** Expansión internacio-nal de sus servicios.	

2.2.2.2 Analizar cadena de valor – Servicio Domésticos Lujan

Dirección General y Recursos Humanos						**Margen del Servicio**
Organización Interna y de Tecnología						
Infraestructura y ambiente						
Abastecimiento						
Marketing y ventas	**Personal de Contacto**	**Soporte Físico y habilidades**	**Presentación**	**Clientes**	**Otros clientes**	
Facebook	Sin Informa-ción, las imágenes presenta-das en re-des son de archivo.	El sitio web esta fuera de servicio, los medios de comunica-ción son a través de re-des sociales o telefónica	Ofrece servicio doméstico y venta de organizadores.	Presencia en zona Lujan	Lanza-miento de talleres para organiza-ción de placard y alacenas	

2.2.3 Mi Cadena de Valor de Servicios

Dirección General y Recursos Humanos						**Margen de Servicio**
Organización Interna y de Tecnología						
Infraestructura y ambiente						
Abastecimiento						
Marketing y ventas Web Instagram YouTube. Twitter Estados en WhatsApp. Facebook	**Personal de contacto** Personal capacitado en limpieza de alfombras, ventanales, tapizados, cortinados.	**Soporte Físico y habilidades** El sitio web ecommerce de Helper permite un contacto directo y permanente con los clientes	**Presentaciones** Variedad de servicios que ninguna otra empresa ofrece en la zona.	**clientes** Tanto el personal como el si-tio web están preparados para recibir comenta-rios de los clientes que le permitirá a la empresa una sinergia para seguir creciendo.	**Otros clientes** A mayor cantidad de clientes satisfe-chos y más pre-sencia en redes con-taremos con mayor cantidad de clien-tes.	

2.3 Comparación de Factores Críticos de Éxito

Un enfoque de factores críticos sirve de pauta para identificar nuestros puntos fuertes, nuestras debilidades y aquellas tareas o secciones que debemos llevar a cabo o modificar para alcanzar las metas. Tomamos en cuenta los siete puntos que propone Hax para poder observar el negocio en su totalidad desde una expectativa muy amplia:

- Infraestructura gerencial.
- Finanzas.
- Gestión de recursos humanos.
- Tecnologías.
- Adquisiciones.
- Fabricación.
- Comercialización y ventas.

Competidor 1 - Servicios Zolvers

Helper Zolvers	Gran Debilidad	Debilidad leve	Equilibrados	Fortaleza Leve	Gran Fortaleza
Infraestructura general					
Finanzas					
Gestión de recursos humanos					
Tecnología					
Adquisiciones					
Fabricación					
Comercialización y ventas					

Competidor 2 - Servicios Domésticos Lujan

Helper Lujan	Gran debilidad	Debilidad leve	Equilibrado	Fortaleza leve	Gran Fortaleza
Infraestructura general					
Finanzas ,.					
Gestión de recursos humanos					
Tecnología					
Adquisiciones					
Fabricación					
Comercialización y ventas					

2.4 Fortalezas y Debilidades

Luego de análisis de nuestros dos principales competidores podemos detectar nuestras fortalezas y nuestras debilidades.

Fortalezas

- Estructura flexible.
- Compromiso de los empleados.
- Reconocimiento y satisfacción por parte de los clientes.
- Personal capacitado.
- Nivel de endeudamiento para el crecimiento controlado.

Debilidades

- Limitación en cantidad de servicios ofrecidos.
- Nivel de ausentismo.
- Competencia desleal en mano de obra informal.

Evaluación general del Negocio	ATRACTIVO ESCASO	ATRACTIVO MEDIO	ATRACTIVO ELEVADO
Evaluación general			

3. F.O.D.A

3.1 Cuadro F.O.D.A

Fortalezas

F1: Estructura Flexible.

F2: Personal Comprometido.

F3: Conocimiento del rubro.

F4: Reconocimiento de los clientes.

Oportunidades

O1: Nuevos Canales de Comunicación

O2: Tarifas Bajas / calidad.

O3: Crecimiento del mercado.

FODA

Debilidades

D1: sustituto del servicio.

D2: Falta de experiencia para grandes volúmenes de cv.

Amenazas

A1: Nuevas agencias.

A2: Competencia desleal.

A3: Industria del Juicio.

A4: Fuerte recesión.

3.2 Análisis F.O.D.A

Fortalezas

F1: Estructura Flexible: debido a los cambios en el contexto de la economía, inflación, social, etc. Las agencias de RRHH deben actuar de manera dinámica para adaptarse al mercado y solicitudes de los clientes.

F2: Personal Comprometido: el nivel de éxito de las agencias de recursos humanos se basa en captar el perfil indicado para cubrir la demanda de los clientes. Debemos ocupamos de tener un personal comprometido, motivado y capacitado para poder brindar el mejor servicio que sea posible y logra cumplir el objetivo.

F3: Conocimiento del rubro: estar al tanto de los exámenes que se deben realizar. Confeccionar un cuestionario simple y claro de acuerdo con el perfil buscado, lo que ayudara a contar con la capacidad de realizar gestiones de búsquedas rápidas.

F4: Reconocimiento de los clientes: para las agencias de RRHH es muy importante la relación y la satisfacción del cliente. Sirve para detectar rápidamente fallas y modificarlas, logrando publicidad para mostrar el nivel de exigencia.

Oportunidades

O1: Nuevos Canales de Comunicación: las páginas webs aceleran el contacto con los clientes y clientes potenciales. Es importante contar con una página actualizada y un buen posicionamiento en los buscadores ya sea por cantidad de visitas y reconocimiento o abonando el servicio de publicidad.

O2: Tarifas Bajas / Calidad: el aumento inflacionario en los costos fijos y la competencia hace que las agencias tengan como política es mantener precios competitivos como estrategia de mercado sin dejar de lado los estándares de calidad.

O3: Crecimiento del mercado: la satisfacción de los clientes en la búsqueda de personal eventual logramos un avance positivo en la cantidad de servicios solicitados.

Debilidades

D1: Sustituto del Servicio: las agencias de recursos humanos tienen competidores desleales como lo es el trabajo informal ò la contratación directa por parte de los clientes. Por otro lado, los problemas económicos desestimulan la generación de nuevos puestos de trabajo.

D2: Falta de experiencia para grandes volúmenes de CV: grandes volúmenes de postulantes cargan a diario sus perfiles en diferentes sitios webs para la postulación a empleos. Si no se cuenta con un gestor adecuado para el filtro de los mismo, se puede perder tiempo valioso o cubrir una vacante con un perfil erróneo.

Amenazas

A1: Nuevas Agencias: con la agilidad y bajo costo que ofrece internet, se observa un aumento en la creación de agencias de búsqueda de recursos humanos. Brindando la oportunidad de crecimiento a competidores ofrecen servicios similares.

A2: Competencia desleal: como agencia de recursos humanos se tiene una estructura de costos fijos que afecta al precio de nuestro servicio, a pesar de que buscamos tener precios competitivos y a su vez brindar un servicio de calidad se nos es muy difícil competir con aquellas personas que trabajan de forma independiente en el mercado informal. Según el UPACP (Unión Personal Auxiliar de Casas Particulares) el 40% de las casas y edificios de lujo del conurbano y puerto madero tiene personal en negro.

A3: Industria del Juicio: se observa un gran aumento en las demandas laborales ya sea a las agencias o a las aseguradoras (ART) lo cual afecta a la generación de nuevos empleos.

A4: Fuerte recesión: Es el caso contrario a la creación de nuevas pymes. Puede darse que nuevas empresas y negocios comiencen a cerrar y la cantidad de nuevos posibles clientes disminuye.

4. Segmentación de Mercado

4.1 Criterios de Segmentación

Existen dos criterios macros de segmentación de mercado:

- Los compuestos por las personas – Mercado de consumo.
- Los compuestos por las organizaciones – Mercado de Negocio.

Nosotros nos vamos a centrar en el mercado de consumo que son los que ocupan las personas que consumen nuestros servicios.

Helper apunta a hombres, mujeres y familias de clases medias y medias altas del conurbano bonaerense, más específicamente en zona oeste y zona norte, que utilicen servicio doméstico.

4.1.1 Rentable

a industria apunta a un mercado de clase media y, clase media alta. En Argentina según la Consultora de Consumo masivo W y Kantar Wordpanel de en un informe publicado en marzo sobre los ingresos por grupo familiar se observan los siguientes datos:

Según la información recolectada por la consultora W y Kantar Wordpanel la clase alta (ABC1) abarca a sólo el 5% de los hogares (600.000 y 1,17 millón de personas), con un ingreso familiar promedio de $ 180.000 mensuales y un piso de $ 90.000. A continuación, se desagrega la clase media alta (C2), con 17% del total (2,04 millones de hogares y 4,70 millones de personas) y un promedio de $ 55.000 mensuales de ingreso (con un piso de $ 33.000); y luego se ubica la clase media típica (C3), con 28% (3,96 millones de hogares y 10,75 millones de personas) que promedia un ingreso familiar de $ 28.000 mensuales, con un piso de $ 18.500.

4.1.2 Accesible

Las empresas buscan ser de fácil acceso para que los clientes puedan comprar, interactuar o conectar con estas fácilmente, sin riesgos o sorpresas desagradables.

Los hilos principales de comunicación con los clientes suelen ser telefónicamente y a través del sitio web, aunque muchas agencias cuentan con una oficina para aquellos que clientes más conservadores ò ignorantes tecnológicos.

Según el director del CACE (Cámara Argentina de Compras Electrónicas) la pandemia impulso al crecimiento del eCommerce, que según el estudio MID que realizó Kantar para CACE, la facturación del comercio electrónico registra un crecimiento del 106% en el primer semestre de 2020 y un 63% más de ventas que en el mismo periodo del 2019. Es evidente que tanto las compras online como los medios de pago electrónico se han consolidado y esto se pudo percibir notoriamente en el Hot Sale, y promete superar ampliamente las expectativas de crecimiento también para el CyberMonday, que se realiza esta semana de noviembre.

El boom del eCommerce durante la pandemia afianzó también el rol de los medios de pago digitales como la manera más segura y eficaz de transaccionar. Lo cierto es que los cuidados protocolares para evitar la propagación del virus llevaron a que los usuarios elijan realizar la mayoría de sus compras de manera virtual, al mismo tiempo que también prefieran digitalizar sus pagos para evitar la manipulación de billetes.

El objetivo es que la experiencia del cliente para acceder a los servicios sea sencilla e intuitiva. Que los pasos ya sean para consultas o compras sean simples y claros.

4.1.3 Medible

A través del Censo 2010 Nacional de Población, Hogares y Viviendas en los 24 partidos correspondientes al Conurbano Bonaerense (Almirante Brown, Avellaneda, Berazategui, Escobar, Esteban Echeverria, Ezeiza, Florencio Varela, General Rodríguez, General San Martin, Hurlingam, Ituzaingó, José C. Paz, La Matanza, Lanús, Lomas de Zamora, Malvinas Argentinas, Marcos Paz, Merlo, Moreno, Morón, Presidente Perón, Quilmes, San Fernando, San Isidro, San Miguel, San Vicente, Tigre, Tres de Febrero y Vicente López) habitan 7.552.415 personas.

Nuestros potenciales clientes conforman las **clases sociales medias y clases sociales altas que suman el 22% de este territorio ósea 1.661.532 habitantes de acuerdo con informes del INDEC** por lo que da un margen medible al segmento del mercado al cual apuntamos.

Durante el 2020 se debería haber llevado adelante el nuevo censo nacional lo cual no fue realizado por el aislamiento social y preventivo. Por lo tanto, no contamos con datos actualizados.

4.1.4 Capacidad de Repuesta

La esencia para dar satisfacción a los clientes es responder rápidamente. Manejando las áreas de cobertura, ser agiles ante consultas ya seas a través de redes sociales, mails, telefónicamente o en persona en los distintos puntos de venta. Tanto sea para consultas, solicitud de servicio o inconvenientes, las cuales deben ser atendidas con mayor rapidez, ya que puede afectar la imagen, antes de ser expuestas en redes sociales. La imagen de una empresa ya sea positiva o negativa queda rápidamente registrada en las redes sociales.

4.2 Variables de Segmentación

En esta sección vamos a identificar aquellos factores que debemos tomar en cuenta para la segmentación del mercado.

4.2.1 Variables Demográficas

Segmentación por Edad

De acuerdo con la segmentación que realizan los mercadólogos se distinguen los siguientes segmentos: recién nacidos (de 0 a 1 año), infantes (de 1 año a 3), niños de 3 a 8 años, los preadolescentes (aproximadamente de 9 a 14 años), adultos jóvenes o generación Milenios abarcan los grupos de edad de 15 a 29 años. Adultos o Generación X de 30 años a 59 años, de los cuales muchos son padres y consideran las opiniones de su familia. La generación de mayores de 60 años en general se concentra en este grupo los sectores más acaudalados y los adultos mayores ò tercera edad superior a los 65/70 años.

Nuestros servicios apuntan a las generaciones de adultos o Generación X los cuales ya dejaron de lado su atención a la moda y lo último en tecnología como lo hacen los Millenians y apuntan más al ámbito familiar y calidad de vida y Generación de adultos con estilos de vida más holgadas.

Resumiendo, los segmentos que utilizan personal doméstico:

- Generación X de 30 años a 59 años, de los cuales muchos son padres y consideran las opiniones de su familia.
- Mayores de 60 años en general se concentra en este grupo los sectores más acaudalados.
- Tercera edad superior a los 65/70 años, donde las limitaciones físicas empiezan a aflorar.

Segmentación por Genero

¿Quiénes gestionan las compras en las familias? Las mujeres suelen llevar adelante las finanzas de las familias. Según un informe de Linio En Argentina las mujeres que más compran tienen entre 25 y 34 años y representan el 41% de las compras femeninas, seguidas por las más jóvenes entre los 18 y 24 años

(20%), por lo que se puede concluir que las millenials representan más del 60% de las compras.

Las mujeres que más compran están en la ciudad de Buenos Aires (37%), seguidas por la provincia de Buenos Aires (24%) y luego por las mujeres en la provincia de Córdoba (11%).

los servicios domestico suelen contratarlo las mujeres ya que son ellas las que acostumbran a llevar adelante esa tarea. Sin embargo, ya no suena extraño que los hombres asuman el rol de la contratación de servicio domésticas, ya que se estima que el 83% de los hombres hace algún tipo de “tareas de la casa” según Dirección General de Estadísticas y Censos de la Ciudad.

Resumiendo, los segmentos por género que utilizan personal doméstico:

- En su mayoría son mujeres, sim embargo, cada vez más los hombres se empiezan a ocupar de dichas tareas.

Segmentación por Ingresos

Helper apunta a un segmento del mercado con ingresos de clase media y media alta, normalmente usuarios con sus necesidades básicas cubiertas. Generalmente personas que consumen servicios como tv por cable, internet, celulares, prepaga, etc. El nivel de ingresos de la clase media y clase alta son superiores a los $80.000.

Segmentación por Ciclo de Vida Familiar

El mercado de Helper apunta a los siguientes siclos de Familias:

- **Personas de mediana edad casada o divorciada sin hijos:**
  - Personas que invierten más dinero en calidad de vida.
  - Personas que disfrutan de tiempo libre.
  - ingresos medios / altos.
  - Ambos miembros de la pareja trabajan.
  - Le interesan viajes y diversión.

- **Personas de mediana edad casados o divorciados con hijos**:
  - Hay esposas que trabajan.
  - Promedio alto de compra de productos duraderos.
  - Adquieren electrodomésticos para el hogar.
  - Le interesan los productos nuevos.

- **Personas mayores casadas o solteras:**
  - están más satisfechos con su situación económica.
  - Compran aparatos médicos, productos que ayudan a la salud, al sueño y a la digestión.

4.2.2 Variables Psicográficas

La segmentación psicográfica es la segmentación del mercado de acuerdo con las siguientes variables:

personalidad: refleja los rasgos, actitudes y hábitos de una persona. El perfil de nuestro consumidor es más conservador, que prefiere tener bajo control los distintos aspectos de los quehaceres domésticos. Por otro lado, a algunas personas tener personal doméstico les da la sensación de tener “otro nivel”.

motivos: los motivos emocionales a los que apuntamos es que las personas que consumen nuestros servicios sientan que nuestro personal es como de su familia y que van a realizar el trabajo con tanta dedicación como si fuera su propia casa. Y que al llegar a su hogar luego de un día agotador los espera su hogar impecable solo para disfrutar de su tiempo libre.

Estilo de vida: la segmentación por estilo de vida divide a las personas en grupos con base en la forma en que pasan el tiempo, la importancia de las cosas que las rodean, sus creencias y características socioeconómicas, como el ingreso y la educación Nuestros clientes son aquellas personas y familias que pasan varias horas al día fuera de casa, sobre todo trabajando. Sea el motivo por el que estén ocupados diariamente, el tiempo libre es escaso y valioso, por lo que necesitan alguien en que confiar los quehaceres domésticos y los cuidados de la casa.

Geo demografía: la segmentación geo demográfica agrupa a los clientes potenciales en categorías de estilo de vida de su vecindario. Combina las segmentaciones geográficas, de nuestros clientes más atractivos están ubicados en barrios cerrados y countries, de clase alta. Donde el servicio doméstico forma parte de su vida y estatus.

4.2.3 Variables Geográficas

Para tener una capacidad de respuesta ágil, nuestro personal vive en la misma zona de cobertura donde brindamos el servicio. Ciudades y barrios urbanos con diferentes tipos de medios de transporte. La temperatura en Moreno, por ejemplo, tiene un promedio anual es de 16.2 C y las precipitaciones 1033 mm al año. Lo que significa que sin importar la época del año suele ser un clima estable.

4.2.4 Variables por Tasa de Uso

Las categorías varían según el servicio, pero probablemente se incluya alguna de las siguientes combinaciones: usuarios anteriores, usuarios potenciales, usuarios por primera vez, usuarios irregulares, usuarios medianos y usuarios frecuentes. Como sabemos los usuarios frecuentes representan un porcentaje considerable de todas las ventas de los servicios ya que son la masa que representa nuestros ingresos para sostener nuestra estructura. Los clientes regulares suelen ser diarios, semanales, quincenales y por eso es importante su opinión sobre nuestro servicio para poder detectar rápidamente fallas y errores.

Los programas de frecuencia/lealtad, se han diseñado programas de lealtad que premian al segmento de los usuarios considerables con descuentos especiales en servicios seleccionados.

4.2.5 Variables por Beneficios

La segmentación por beneficios es el proceso de agrupar a los clientes en segmentos de mercado con base en los beneficios que buscan del servicio. Agrupa a los clientes potenciales con base en sus necesidades o deseos. Nuestra empresa ofrece además de servicio doméstico, limpieza de alfombras, limpiezas de cortinas, limpieza de tapizados, ventanales entre otros.

4.3 Mercado Meta

Un mercado meta es un grupo de personas para los cuales una organización diseña, implementa y mantiene una mezcla de marketing creada para satisfacer las necesidades de dicho grupo, dando como resultado intercambios mutuamente satisfactorios.

Las tres estrategias generales para seleccionar a los mercados meta no diferenciada (adopta la filosofía de un mercado masivo y lo considera un gran mercado sin segmentos individuales. La empresa utiliza una mezcla de marketing para todo el mercado), concentrada (una empresa selecciona un nicho de mercado (un segmento de mercado) para dirigir sus actividades de marketing.) y de segmentos múltiples) (Una empresa que elige servir a dos o más segmentos de mercado bien definidos y desarrolla una mezcla de marketing distintiva para cada uno).

Para nuestro emprendimiento utilizaremos la estrategia de *mercado concentrado,* ya que focalizar los recursos y satisfacer las necesidades de un segmento de mercado muy definido es más productivo que extender los recursos entre varios segmentos:

Pasando el claro, el perfil del mercado concentrado de Helper es:

- Clientes con ubicación geográfica de zona oeste y zona norte. Centros urbanos de Moreno, La Reja, Fco. Álvarez, Gral Rodríguez y Pilar entre otros. Apuntando a aquellos que viven en barrios cerrados y countries.
- Familias con ingresos correspondientes a clase media / alta y clases altas. Jóvenes adultos, personas de mediana edad, adultos mayores. Personas profesionales, que valoran su tiempo libre y el cuidado del hogar.

5. Estrategia de Negocio

5.1 Matriz Atractivo de la Industria / Fortaleza del Negocio

Analizando los puntos anteriores, resumimos estos tomando en cuenta 2 factores críticos: el atractivo de la industria y la fortaleza del negocio. Por esto, nos ubicamos indicando el recuadro en celeste, dentro de la siguiente matriz:

Estrategias Genéricas

Atractivo de la Industria

Fortaleza del negocio

	Alta	Media	Baja
Alta	Crecimiento Búsqueda de predominio Maximizar la inversión	Identificar segmentos de crecimiento Invertir fuertemente Mantener la posición en otras partes.	Mantener la posición general Buscar flujo de fondos Invertir a nivel de mantenimiento
Media	Evaluar potencial para el liderazgo a través de la segmentación Identificar las debilidades Fortalezas	**Identificar los segmentos de crecimiento** **Especializarse** **Invertir en forma selectiva**	Eliminar líneas Minimizar la inversión Posicionarse para reestructuración.
Baja	Especializarse Buscar nichos Considerar adquisiciones	Especializarse Buscar nichos Considerar la salida	Confiar en la calidad de estadista del líder Atacar los ganadores de efectivo de los competidores. Salida y abandono oportuno

Análisis de la Industria

Actual Futuro	Muy Poco Atractivo	Poco Atractiv0	Neutro	Atractivo	Muy Atractivo
Análisis de la Industria					

Análisis del Negocio

Evaluación general del Negocio	ATRACTIVO ESCASO	ATRACTIVO MEDIO	ATRACTIVO ELEVADO
Evaluación general			

5.2 Programas de Generación de Acción

Cada programa general de acción está respaldado por un conjunto de programas específicos de acción que ayudan a proporcionar un sentido de concreción al trabajo gerencial estratégico. Los programas específicos de acción son tareas tangibles a corto plazo que pueden ser identificados, controlados y evaluados con precisión.

5.2.1 Programas de Administración

Programa Estratégico de Administración
Descripción
El programa estratégico de administración es el de llevar adelante aquellos procesos para las tareas de la empresa en los aspectos de recursos humanos, costos fijos, costos variables, tesorería, cobranzas, contable.
Responsable
Encargado Administrativo, estudio contable
Indicadores de Gestión
ventas concretadas vs ventas presupuestadas. Gastos (fijos) realizados vs gastos presupuestados. Análisis de Ganancias Análisis de perdidas (sueldos, costo de promociones o resarcimientos para clientes) Gastos bancarios, comisiones de créditos para inversión en nuevos servicios. Balance anual.
Fecha de Inicio – Finalización del programa
enero 2021 – diciembre 2023

5.2.2 Programas de Servicios

Programa General de Servicios
Descripción
Programa estratégico para los servicios tomados. Estrategias de precios, promociones, organización de los trabajadores.
Responsable
Gerente General Encargado de servicios
Indicadores de Gestión
Procesamiento de solicitudes de servicios. Procesamiento de servicios brindados. Procesamiento de datos de pagos a través de tarjetas. Coordinación y organización del personal para la realización de los servicios solicitados. Supervisa los servicios brindados
Fecha de Inicio – Finalización del programa
Enero 2021 – diciembre 2023

5.2.3 Programa Estratégico de Marketing

Programa Estratégico de Marketing
Descripción
El rol de este departamento es definirá estrategias de nuevos servicios y precios, descuentos y/o promociones. Analizando e implementando estrategias para captar nuevos mercados.
Responsable
Encargado Comercial
Indicadores de Gestión
Determinar sobre qué servicios se va a aplicar la promoción. Lealtad hacia la marca esperada frente a la lealtad real. Cantidad de visitas al sitio web esperadas frente al número de visitas al sitio web real. Nivel de satisfacción del cliente esperada frente al nivel de satisfacción del cliente real.
Fecha de Inicio – Finalización del programa
enero 20121– diciembre 2023

5.2.4 Programa Estratégico Comercial

Programa Estratégico Comercial
Descripción
Programa estratégico comercial para la coordinación de servicios y compromisos adquiridos para mantener una relación positiva con los clientes.
Responsable
Gerente General Encargado de Servicios
Indicadores de Gestión
Atención al cliente través del sitio web para la toma de servicios. Atención a través del sitio web para la toma de consultas. Respuesta a solicitudes de presupuesto. Atención al cliente telefónicamente y oficina. Aumento de servicios contratados con respecto al mes anterior.
Fecha de Inicio – Finalización del programa
Enero 2021 – diciembre 2023

5.2.5 Programa Estratégico Tecnológico

Programa Estratégico Tecnológico
Descripción
Programa estratégico tecnológico consiste en la actualización, mantenimiento, actualización de página web, procesamiento de información recepción ya sea de clientes o postulantes.
Responsable
Encargado de Tecnología
Indicadores de Gestión
Creación de perfiles de clientes y personal administrativo. Cantidad de quejas por parte de usuarios menor respecto al mes anterior. Tiempo de respuesta estable de los sistemas de la empresa Diseños de página ágil e intuitiva para que sea atractiva al cliente. Software. Hardware. Telecomunicaciones.
Fecha de Inicio – Finalización del programa
Febrero 2021 – enero 2023

5.2.6 Relación matriz FODA y programas estratégicos

	Administración	Servicios	Marketing	Comercial	Tecnológico
Fortalezas					
Estructura Flexible		•	•		
Personal Comprometido	•	•			•
Conocimiento del rubro		•		•	•
Reconocimiento de los clientes		•	•		
Oportunidades					
Nuevos canales de comunicación				•	•
Tarifas bajas / calidad		•	•	•	
Crecimiento del mercado				•	•
Debilidades					
Sustituto del servicio				•	
Falta de experiencia en procesar grandes volúmenes de datos.	•	•			
Amenazas					
Nuevas agencias				•	•
Competencia desleal				•	•
Industria del juicio.				•	•
Fuerte recesión				•	

5.3 Programas Específicos de Acción

Cada programa general de acción está respaldado por un conjunto de programas específicos de acción que ayudan a proporcionar un sentido de concreción al trabajo general estratégico. Los programas específicos de acción son tareas tangibles a corto plazo que pueden ser identificadas, controladas y evaluadas con precisión.

5.3.1 Programa Generales

Programa General	Programas específicos
Estrategia de Administración	• Administración de compras. • Administración de Finanzas • Administración contable. • Administración de Cobranzas. • Administración de RRHH • Administración de Pagos
Estrategia de Comercial	• Gestión de venta • Estrategia de posventa
Estrategia de Marketing	• Promociones / Descuentos.
Estrategia de servicios	• Gestión de solicitudes vía web. • Gestión de presupuestos.
Estrategia de Tecnología	• Estrategia de software. • Estrategia de Hardware. • Mantenimiento / soporte.

5.3.2 Programa Específicos de Administración

Programas Específicos de Compras
Descripción
Realiza los procedimientos para la compra de materiales para la realización de las tareas. Asegurarse que este disponibles la cantidad de herramientas necesaria para ofrecer los servicios. Buscar precios competitivos, solicitar presupuestos, la financiación más conveniente y garantías.
Tareas
• Buscar proveedores ya sean de producción nacional o maquinaria importada. • Hacer una investigación de mercado para conseguir un herramientas y elementos de calidad. • Gestor de precio, buscar precios competitivos para no vernos obligados a subir el costo de los servicios. • Modo de financiación, tarjetas de crédito, cheques a plazo (30, 60, 90 días) • El caso de herramientas que cuente con garantía.
Responsable
Gerente General
Equipo de Trabajo
Asistente Administrativo
Indicadores de Gestión
• Solicitud de presupuesto. • Rechazo ò aceptación del presupuesto. • Orden de compra. • Factura.
Fecha de Inicio – Finalización del programa
enero 2021 – diciembre 2023

Programas Específicos de Finanzas
Descripción
Las finanzas corresponden al movimiento de dinero dentro de la administración de la empresa ya sea para pagar productos, servicios, aportes patronales, etc. Los movimientos contables (presupuestos) y de tesorería (caja y bancos).
Tareas
• Se ocupa de armar el presupuesto financiero, controlarlo y detectar ítems que deban modificarse en caso de que no se cumpla. • Administración de los pagos, plazos y medios. • Gestiones bancarias, depósitos bancarios, asegurar la existencia de fondos en caso de que hayamos dado un cheque. • Carpeta crediticia con el banco. Mantener semestralmente el legajo bancario.
Responsable
Administrativo
Equipo de Trabajo
Asistente Administrativo
Indicadores de Gestión
• Balance. • Pagos. • Orden de pago. • Depósitos bancarios. • Transferencias. • Estados de cuentas bancarias. • Asientos contables.
Fecha de Inicio – Finalización del programa
enero 2021 – diciembre 2023

Programas Específicos de Contable
Descripción
La administración contable se encarga de llevar el registro diario de cada una de las actividades de la empresa. es muy importante tener un registro claro, confiable, al cual se pueda acceder rápidamente para tener un informe de cómo se encuentra económica y financieramente. Su tarea es llevar a cabo el presupuesto.
Tareas
• Llevar adelante el cumplimiento del presupuesto. • Realizar los asientos contables y cierres. • Listados de facturación. • Impuestos, IVA, ingresos brutos, aportes. • Conciliaciones bancarias. • Liquidación de 931, declaraciones juradas. • Estados financieros.
Responsable
Administrativo
Equipo de Trabajo
Asistente administrativo.
Indicadores de Gestión
• Estados de cuentas bancarias. • Asientos contables. • Informes de Ingresos. • Informes de Egresos • Saldos totales.
Fecha de Inicio – Finalización del programa
enero 2021 – diciembre 2023

Programas Específicos de Cobranzas
Descripción
Realiza la gestión de cobranzas. Su función consiste en que las cuentas corrientes de los clientes estén al día. Asegurarse que este actualizadas las partidas pendientes de los servicios brindados al cliente. Registra pagos y emite recibos.
Tareas
• Emite partidas pendientes de la cuenta de los clientes. • Emite recibos. • Carga pagos. • Facturación. • Contacto con el cliente solicitando servicios pendientes de pagos. • Verifica depósitos realizados por parte de los clientes. • Bloqueo de clientes. • Habilitar pedidos de servicio. • Habilitar créditos.
Responsable
Administrativo
Indicadores de Gestión
• Partidas pendientes. • Listados de cobranzas. • Asientos de cobros. • Listado de deuda por antigüedad. • Estadísticas de clientes.
Fecha de Inicio – Finalización del programa
enero 2021 – diciembre 2023

Programas Específicos de RRHH
Descripción
La liquidación de haberes de la empresa es de manera tercerizada, por lo que le enviamos la información correspondiente a cada persona y un estudio se ocupa de la liquidación.
Tareas
• Informe de horas trabajadas por empleado. • Presentación de certificados en caso de ausencia justificada. • Valores de las horas según acuerdo gremial. • Selección de personal. • Plan de capacitación para el personal. • Beneficios por logros obtenidos. • Beneficios en casos de festejos, aniversarios, casamientos, nacimientos, etc.
Responsable
Administrativo – Estudio Contable
Equipo de Trabajo
Asistente Administrativo
Indicadores de Gestión
• Sumatoria de horas según costo de valor hora. Lunes a viernes al 100% sábados al 120%. • Informe de ausentismo.
Fecha de Inicio – Finalización del programa
Enero 2021 – diciembre 2023

Programas Específicos de Pagos
Descripción
Se realizan las operaciones referentes a la cancelación de las obligaciones contraídas.
Tareas
• Recepción y control de la documentación de pagos a realizar contra las órdenes de compra, factura, remitos y conformidad de recepción • Emisión de órdenes de pago • Se envía la documentación a Finanzas. Finanzas controla con el presupuesto financiero que se posea dinero para realizar el pago y lo envía a Tesorería • Autoriza el Pago y lo envía al Encargado de Pagos • Se confecciona la documentación para realizar el pago o se concreta en efectivo y se guardan los comprobantes correspondientes. • Se registran contablemente los pagos efectuados
Responsable
Administrativo
Indicadores de Gestión
• Ordenes de pagos emitidas. • Comprobantes de pago. • Estado de endeudamiento. • Plazos pactados de pago.
Fecha de Inicio – Finalización del programa
Enero 2021 – Diciembre 2023

5.3.3 Programas Específicos de Comercial

Programas Específicos de Gestión de Venta
Descripción
El objetivo es llevar adelante una gestión en ventas que permita general más ganancias. Por lo que para vender más servicios es necesario tener una organización prolija, ya que, si tomamos muchos servicios, pero se superponen los horarios y el personal no da a vasto será un grave error.
Tareas

• Planificación de los servicios tomados y seguimientos de estos.
• Mantener las listas de precios según el servicio, días y horarios actualizadas.
• Atender al público en el punto de venta y telefónicamente.
• Atender al público a través de las consultas y solicitudes a través del sitio web.
• Realizar presupuestos.
• Genera ficha cliente.
• Revisa si el cliente tiene deuda.
• Organización del personal según tarea a realizar.
• Organización del personal de acuerdo con la zona.
• Ordenes de trabajo diarias y por personal.
• Establecer en que consiste claramente el trabajo a realizar ya que de acuerdo con la tarea es el valor de la hora. Esto es muy importante para que no haya malentendidos con los clientes.
• Establecer formas de pago:
  ○ A través del sitio web: tarjeta de crédito únicamente.
  ○ En punto de venta: efectivo, débito o tarjeta de crédito.
• Posteo en nuestro canal de YouTube videos con la experiencia de clientes satisfechos con el servicio.

Responsable
Ventas
Indicadores de Gestión
• Ordenes de trabajos. • Nivel de facturación.
Fecha de Inicio – Finalización del programa
enero 2021 – diciembre 2023

Programas Específicos de Posventa
Descripción
Servicio y atención que se le brinda al cliente luego de la tarea brindada. Esto se debe como garantía del servicio, atención a comentarios, sugerencias para captar la intención de consumir de nuevo nuestro servicio. Ofreciendo servicios o promociones como resarcimiento en caso de insatisfacción del cliente.
Tareas
• Responder ante quejas. • Solución de inconvenientes o retrasos. • Mantenimiento de una comunicación clara con el cliente sobre el tipo de servicio contratado. • Identificar oportunidades de mejoras / sugerencias. • Ofrecer descuentos o promociones como compensación.
Responsable
Ventas
Indicadores de Gestión
• Informes sobre clientes. • Estadísticas sobre inconvenientes y empleado a cargo
Fecha de Inicio – Finalización del programa
Enero 2021- diciembre 2023

5.3.4 Programa Específicos de Marketing

Programas Específicos de Promociones / Descuentos
Descripción
Como estrategia para ganar mercado utilizamos promociones como tácticas de ventas. La publicidad de promociones nos permite ganar nuevos clientes, mantener los actuales y reconquistar a aquellos que se alejaron.
Tareas
• Realizar campañas publicidad en la vía pública. • Enviar a nuestra cartera de clientes de los cuales contamos con correo electrónico las ofertas, noticias, novedades vía correo electrónico. • Publicidad en medios especializados. • Publicidad en medios de comunicación. • Contactar clientes vía telefónica, sin descuidar el horario y días de llamado para que sea contraproducente. • Establecer estrategias con tarjetas de crédito bancarias con planes de cuotas sin interés. • Promociones por paquetes de horas: servicios gratis adicionales ò horas extras sin abonar excedente. • Seguimiento de variables de tasas de uso de nuestros clientes. • Análisis de competidores. • Definición de estrategia de precio. • Análisis de sensibilidad.
Responsable
Gerente General
Equipo de Trabajo
Ventas
Indicadores de Gestión
• Informes sobre clientes inactivos / clientes recuperados. • Altas de clientes. • Nivel de respuestas vía correo electrónico / telefónicamente o en nuestros puntos de ventas.
Fecha de Inicio – Finalización del programa
Enero 2021 – diciembre 2023

5.3.4 Programa Específicos de Servicios

Programas Específicos Consultas vía Web
Descripción
Como estrategia para ganar mercado utilizaremos el sitio web como herramienta de acercamiento a clientes ya sea en cualquiera de sus categorías.
Tareas

• Análisis de las consultas recibidas:
  ○ Clasificación del consultante (nuevo, inactivo, activo)
  ○ Clasificación por tipo de servicio.
  ○ Ubicación.
 	
• Responder rápidamente, siendo claro en los términos, condiciones y costos.
• Vocabulario simple, siempre a disposición de cliente.
• Solicitar datos del consultante para realizar un seguimiento.
• Transmitir confianza y poner a disposición referencias del personal.

Responsable
Ventas
Indicadores de Gestión
• Informes sobre clientes inactivos / clientes recuperados. • Altas de clientes. • Nivel de respuestas vía correo electrónico.
Fecha de Inicio – Finalización del programa
Enero 2020 – diciembre 2023

Programas Específicos de Presupuestos
Descripción
Como estrategia para ganar mercado utilizaremos el sitio web como herramienta de acercamiento a clientes ya sea en cualquiera de sus categorías. Realizamos seguimiento y control sobre los presupuestos enviados para detectar prontamente falencias y ofrecer promociones o descuentos.
Tareas

• La gestión y control de presupuestos es una tarea importante para la captación de nuevos consumidores. Ya sean clientes regulares que buscan nuevos servicios o cliente nuevos.
• Es muy importante antes de enviar un presupuesto hacernos de la mayor cantidad posible de información del consultante.
  ○ Si es nuevo consumidor de este tipo de servicio.
  ○ Si ya tiene experiencia y quiere cambiar su proveedor.
  ○ Tratar de detectar si es un potencial cliente genuino.
• Debemos ser muy atentos y claros ya que es el primer acercamiento al cliente y según nuestra respuesta será la imagen que tendrá de nosotros.
• Nuestros precios son fijos y competitivos en, relación con el precio del mercado y han sido establecidos luego de análisis, costos y beneficios.
• Siempre dejar la puerta abierta a algún tipo de promoción o descuento.

Responsable
ventas
Indicadores de Gestión
• presupuestos. • Altas de clientes. • Nivel de respuestas vía correo electrónico.
Fecha de Inicio – Finalización del programa
Enero 2021 – diciembre 2023

5.2.5 Programa Específicos Tecnológico

Programas Específicos de Estrategia de Software
Descripción
Como estrategia para tener un mayor control y organización en la empresa optamos por un software a la medida de las necesidades. El sistema apunta a facilitar la comunicación y el acercamiento al cliente: • Agilizando la contratación de servicio doméstico de manera on-line. • Solicitudes de presupuestos y consultas según las categorías de servicios que prefiera a través de una página. • Perfil de cliente donde podrá acceder a su historial de compras y programas servicios. Funciones de control para el empleador: • Control de su personal. Es de fácil uso y concede llevar a cabo las tareas relacionadas con el control de la nómina. • Control de asistencia del personal. • Calcular valor de horas trabajadas por tareas realizadas según su categoría. • Permitiendo llevar una agenda de trabajos programados. • Posibilitando la emisión de un reporte con el detalle de servicios brindados por cada empleado y estadísticas por satisfacción del servicio. • El sistema busca manejar de manera segura la información cotidiana sobre los servicios tomados, empleado a cargo de llevar a cabo dicha tarea, horas programadas y costos de este. Aplicación para postulantes: • Completar formulario para aplicar a puestos.
Tareas
• Planificación • Análisis de requisitos. • Diseños del sistema. • Diseño de persistencia. • Diseños del programa. • Codificación. • Pruebas. • Implementación.

- Mantenimiento y actualización.
- Creación de perfiles.
- Antivirus.
- Licencias de sistemas operativos Windows.

Responsable

Desarrollador Fullstack

Equipo de Trabajo

Analistas de datos - Desarrollador Fullstack

Indicadores de Gestión

- Hitos de avance.
- Informes de requisitos.
- Documentación del sistema
- Manual de usuario.

Fecha de Inicio – Finalización del programa

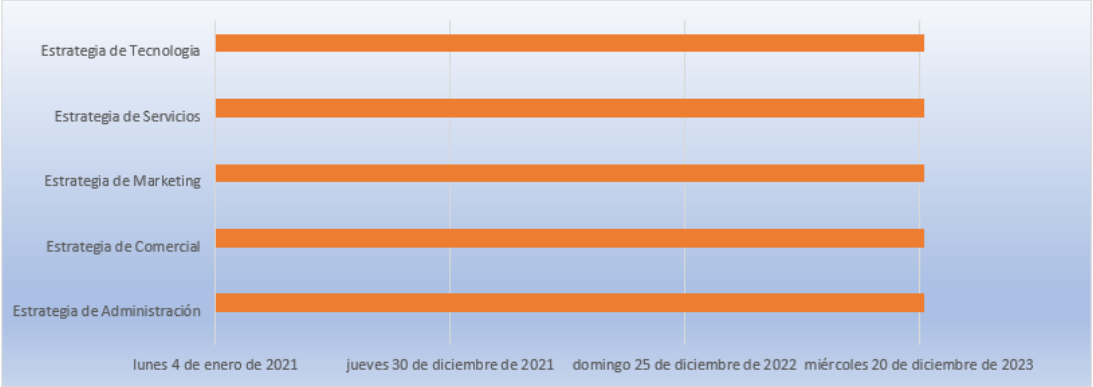
enero 2021 – diciembre 2023

Programas Específico de Estrategia de Hardware
Descripción
Esta área se encarga de la administración de los sistemas operativos y al soporte a los usuarios para garantizar el funcionamiento de los equipos.
Tareas
• instalación y mantenimiento del servidor. • Instalación y mantenimiento de impresoras. • Informar y solicitar nuevos equipamientos. • Protocolos de seguridad (disco espejo). • Mantenimiento de los equipos. • Detección y resolución de averías. • Evaluar necesidades de recursos. Como memoria, discos rígidos
Responsable
Analista de datos.
Indicadores de Gestión
• Informes de estados. • Solicitudes de equipamientos. • Solicitudes de repuestos. • Baja en la tasa de errores debido a fallas de hardware.
Fecha de Inicio – Finalización del programa
enero 2021 – diciembre 2023

Programas Específicos de Mantenimiento y Soporte
Descripción
Como estrategia para evitar inconvenientes en el funcionamiento contamos con personal de mantenimiento y soporte. La cual garantiza el funcionamiento y la pronta reparación en caso de algún inconveniente.
Tareas
• Mantenimientos y actualización de las terminales que utiliza el personal de la empresa. • Back up y restauración de la base de datos. • Mantenimiento de la red de la empresa. • Actualización de licencias de sistemas operativos. • Instalación y configuración de servidores. • Actualización de licencias antivirus. • Mantenimiento de impresoras. • Altas y bajas de usuarios. • Mantenimiento de servicio de correos electrónicos. .
Responsable
Analista de datos
Equipo de Trabajo
Analista de datos – programador fullstack
Indicadores de Gestión
• presupuestos. • Solicitudes de equipos y repuestos. • Solicitud de cartuchos para impresoras.
Fecha de Inicio – Finalización del programa
enero 2021 – diciembre 2023

5.4 Gantt

Nombre de Tarea	Duracion	Comienzo	Finalizacion
Estrategia de Administración	780	lunes 4 de enero de 2021	viernes 29 de diciembre de 2023
Estrategia de Comercial	780	lunes 4 de enero de 2021	viernes 29 de diciembre de 2023
Estrategia de Marketing	780	lunes 4 de enero de 2021	viernes 29 de diciembre de 2023
Estrategia de Servicios	780	lunes 4 de enero de 2021	viernes 29 de diciembre de 2023
Estrategia de Tecnologia	780	lunes 4 de enero de 2021	viernes 29 de diciembre de 2023



6. Plan de Marketing

6.1 Estrategia de Servicios

Un servicio se puede definir como todo aquello, propicio o adverso, que una persona recibe en un intercambio. Puede ser un bien tangible como un par de zapatos, un servicio como un corte de cabello, una idea como “No tire basura” o cualquier combinación de estos tres.

Un servicio es el resultado de aplicar esfuerzos humanos o mecanismos a personas u objetos. Los servicios incluyen una acción, desempeño o esfuerzo que no se pueden poseer físicamente. Una población que envejece necesita enfermeras, cuidados médicos a domicilio, terapeutas físicos y trabajadores sociales. Las familias con dos personas que trabajan necesitan servicio de cuidado infantil, limpieza de hogares y jardinería.

La estrategia de servicios cuenta con la forma y metodologías para estar cada vez más cerca de nuestros clientes y, además, atraer clientes potenciales. Con esto se busca un incremento de solicitudes de servicios y que nuestros clientes siempre opten por nosotros y no a nuestros competidores.

6.1.1 Descripción de los servicios

Los servicios se pueden clasificar como de negocios (industriales) o de consumo, según las intenciones del comprador. La diferencia clave entre ambos tipos es el uso que se les da. Si el propósito es de negocios, el servicio se clasifica como de negocios o industrial. Un servicio de negocios se utiliza para fabricar otros bienes o servicios, para facilitar las operaciones de una organización o para revenderlo a otros clientes. Un servicio de consumo se compra para satisfacer los deseos personales de un individuo.

En el caso de Helper, el servicio es de consumo, ya que se compra para satisfacer los deseos personales de un individuo. Como se indicó en apartados anteriores, el segmento de mercado meta apuntado busca satisfacer sus necesidades de sobresalir al círculo social y buscar satisfacer su imagen de sí mismo.

Helper se encuentra en la clasificación de servicios de consumo en el segmento **servicios de conveniencia** es un servicio relativamente barato que merece poco esfuerzo de compra; es decir, el consumidor no está dispuesto a realizar una búsqueda exhaustiva para conseguir dicho servicio y **servicios buscados,** por lo general, es más costoso que uno de conveniencia. Los consumidores casi siempre compran un servicio buscado después de comparar varias marcas o tiendas en términos de estilo, practicidad, precio y compatibilidad con su estilo de vida.

Tipos de servicios:

- Servicio doméstico de limpieza Interior.
- Servicio doméstico de limpieza exterior.
- Limpieza de alfombras.
- Limpieza de ventanales.
- Limpieza de cortinados.
- Limpieza de tapizados.

6.1.2 Características del servicio

Helper ofrece servicios domésticos e incorporo maquinarias para los servicios de limpieza de alfombras, tapizados, limpieza a presión de agua, limpieza de cortinados y ventanales.

Los servicios tienen cuatro características únicas que los distinguen de los bienes. Los servicios son intangibles (no pueden tocarse, verse, probarse, escucharse o sentirse en la misma forma que puede hacerse con los bienes), inseparables (los consumidores deben estar presentes durante la producción de servicios), heterogéneos (variabilidad de insumos y resultados, tienden a ser menos estandarizados y uniformes que los bienes) y perecederos. En el caso de Helper los servicios son intangibles, sin embargo, a través de los comentarios de nuestros clientes cuentan sus opiniones y experiencias de nuestros servicios.

Nuestro personal recibe capacitación constante y se los instruye en la manera de trabajar. Buscamos la estandarización en el servicio, para así, sin importar que personal este asignado a tal tarea el cliente siempre reciba la misma calidad

del servicio. Dicho servicio cumple con la condición de perecedera, lo cual significa que no pueden ser guardados, almacenados o inventariados.

6.1.3 Calidad del servicio

En esta sección aquellas características que poseen nuestro servicio que hacen que los consumidores los prefieran antes que la competencia:

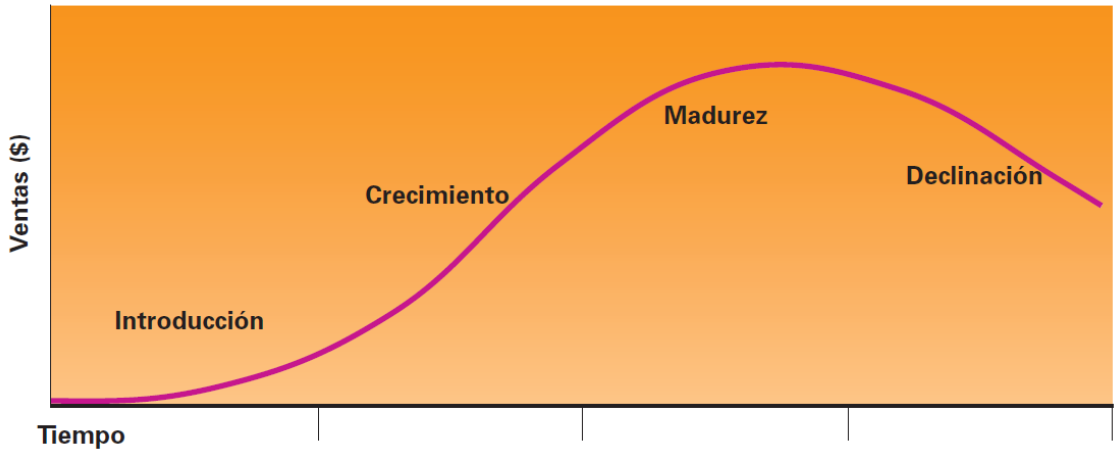
- **Confiabilidad**: los servicios se ofrecen de manera confiable, precisa y uniforme con personal capacitado. Nuestro objetivo es brindar el mejor servicio que nos sea posible desde el primer momento.
- **Capacidad de Respuesta**: la capacidad de brindar un servicio rápido ya sea en respuesta a un llamado telefónico de un cliente, una solicitud de presupuesto y que nuestro personal cumpla estrictamente el día y horario estipulado para brindar el servicio.
- **Seguridad**: nuestro personal está altamente capacitado y estará disponible para el cliente ante cualquier consulta o inquietud. Es muy importante que el cliente se sienta seguro ante la presencia de nuestros empleados ya que nuestro trabajo se lleva adelante en su hogar, por lo tanto, la cortesía, los buenos modales y el respeto son esenciales.
- **Empatía**: llevamos adelante reportes de los servicios brindados y estadísticas de nuestros clientes. Buscamos reconocerlos, individualizarlos, saber sus nombres y sus requerimientos específicos.
- **Tangibles**: por último, podemos mencionar lo tangible de nuestro servicio que implica toda la vestimenta del personal y maquinarias para limpiezas específicas.

6.1.4 Líneas – mezclas

- **Línea de productos** Grupo de productos individuales estrechamente relacionados.
- **Mezcla de productos** Todos los artículos que una organización comercializa.
- **Ancho (o amplitud) de la mezcla de productos** Número de líneas de productos que ofrece una organización.
- **Profundidad de la línea de productos** Número de artículos en una línea de productos.

	Ancho de mezcla de Servicios			
	Tipo de Limpieza	**Limpieza Profunda**	**Limpieza interior**	**Limpieza Exterior**
Profundidad de las líneas de Servicios	Limpieza de Cortinas	Servicio de limpieza profunda. (Interior y Exterior) Incluye todos los servicios.	Servicio mensual.	
	Limpieza de Alfombras.		Servicio mensual.	
	Limpieza de Ventanales		Servicio mensual.	
	Limpieza Domestica		Mantenimiento de limpieza.	Limpieza veredas, Entradas, recibidor, quinchos, baños externos, etc.

6.1.5 Ciclo de vida del Servicio



Etapa de Madurez

El periodo en el cual las ventas aumenta a una tasa cada vez menor señala el inicio de la etapa de madurez del ciclo de vida. No pueden atraer nuevos usuarios de forma indefinida y tarde o temprano el mercado de acerca a una saturación. En este, por lo general, es la etapa más larga del ciclo de vida del producto.

En esta etapa del ciclo de vida, la competencia se vuelve más feroz y, por tanto, la publicidad persuasiva y de recordatorio se enfatiza con mayor intensidad. La promoción de ventas regresa al enfoque cuando los vendedores de los servicios tratan de aumentar su participación de mercado. Mantenemos un precio competitivo sin decaer en la calidad es la manera de mantener la cartera de clientes.

	MADUREZ
SERVICIOS	La empresa cuenta con una variedad de servicios ya sean básico como aquellos de limpieza especifica (limpieza de alfombras, tapizados, cortinados y grandes ventanales) Siempre atentos a las posibilidades de introducir nuevos servicios.
PLAZA (DISTRIBUCIÓN)	Zona de cobertura de servicio limitada. Nos centramos en zona oeste y Pilar.
PROMOCIÓN	La empresa se esfuerza por retener y fidelizar a sus clientes a través de promociones y acciones específicas de marketing.
PRECIO	En ocasiones disminuyen los precios los servicios de Helper debido a que sus competidores bajan los precios.

6.1.6 Estrategia de Marca

6.1.6.1 Definir Marca

6.1.6.2 Características de la marca

Una de las decisiones más importante que debemos tomar es la marca que le vamos a dar al servicio. Es una tarea muy compleja que no se puede tomar a la ligera:

- Helper, ayudante en inglés, es un nombre adecuado ya que en una sola palabra transmite quienes somos y que hacemos en la empresa.
- Es un nombre corto, que hace que sea fácil de recordar y pronunciar.

El objetivo es que el cliente cuando escuche la palabra Helper lo asocie con un servicio de limpieza de excelencia.

6.1.6.3 Estrategia de marca

Nuestra marca apunta a lograr que el cliente a través de nuestro nombre, logo, imagen, diseño, en combinación sea identificado por nuestros cliente y potenciales consumidores. El objetivo que se identifica fácilmente que servicio ofrecemos y diferenciarnos de nuestros competidores.

Nuestra estrategia de marca apunta a lo siguiente:

1. Nuestros servicios no son tercerizados, son propios, por ende, vamos a utilizar nuestra marca, somo *marca del fabricante* de los servicios.
2. Nombre común para todos los servicios. Por lo tanto, vamos a utilizar *marca de familia.*

Marca	Helper
Logo	
Eslogan	*"Limpieza Profesional"*

6.2 Estrategia de Precios

En esta etapa se va a describir nuestra estrategia de precios y aquellos elementos que influyen y condicionan nuestros precios.

6.2.1 Análisis de precios

6.2.1.1 análisis de costo

Costos Fijos

- Alquiler local.
- Electricidad
- publicidad
- Internet y teléfono.
- Agua.
- Dominio ecommerce.
- Sueldos de personal.
- Impuestos.
- Estudio contable.
- ABL
- Estudio Jurídico

Costos Variables

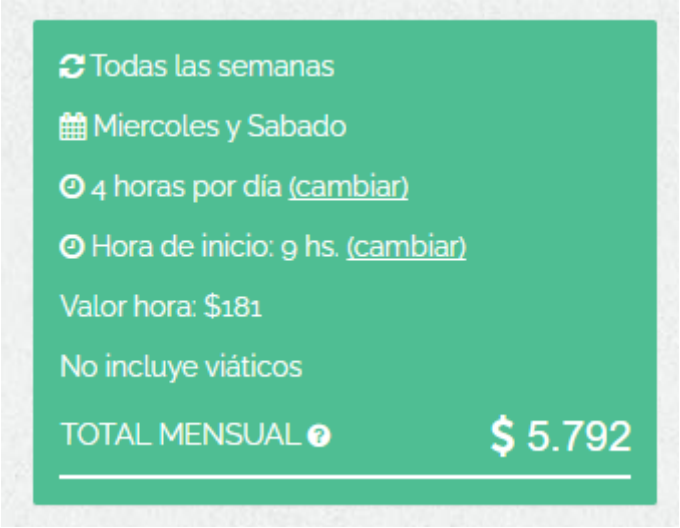
- Gastos de librería.
- Insumos Administrativos (tóner para la impresora, resma de hojas, etc.)
- Viáticos cadete.
- Folletería / promociones.

6.2.1.2 Análisis de Competencia

Competidor Zolver:

Promociona los siguientes precios en su sitio web.

Si solicitamos una cotización mensual de 4 horas semanales, nos informa el siguiente presupuesto en el cual se observa el aumento del costo de la hora.



Competidor Servicio Doméstico Lujan:

Valor de servicio doméstico regular $250. No informa diferenciación.

6.2.1.3 Análisis de Sensibilidad

Los clientes de Helper reaccionan de diferentes maneras de acuerdo con las modificaciones de los precios de nuestros servicios. Un aumento en los precios de nuestros servicios puede afectar al número de servicios contratados.

Para determinar la sensibilidad sobre el precio debemos analizar una serie de ítems como conocimiento de las alternativas, la confianza para hacer comparaciones entre marcas, precio como coste proporción del beneficio total recibido, etc.

Competidores: contamos en la zona con competidores con los cuales es fácil contactarse y hacer consultas sobre sus servicios. Por lo tanto, debemos hacer un esfuerzo para mantener precios atractivos.

Efecto calidad precio: nos esforzamos para mantener el mejor servicio posible y precios atractivos.

Efecto beneficio Final: Nuestro servicio posventa recolecta información de las experiencias de nuestros clientes con los servicios brindados para retroalimentarnos y estar al tanto de posibles fallas.

Efecto Justicia: en determinados casos se puede beneficiar al cliente con algún tipo de descuento o bonificación.

6.2.3 Estrategia de Precios

Nuestra estrategia de precios se encuentra posicionada en el sector neutro o *statu quo* que consiste en la igualación de precios con la competencia o fijación de precios de tarifa actual, ósea que el precio sería igual que la competencia o muy cercano.

Ubicarnos en el rango neutro de los precios nos favorece en el igualamiento con la competencia y nos establece a largo plazo.

Tipo de Cliente	Limpieza Domestica	Limpieza de Cortinas	Limpieza de alfombras	Limpieza de Ventanales
Cliente Eventual	$250	$375	$520 ($m^2$)	$375
Cliente diario con cuenta corriente.	$225	$337,50	$468 ($m^2$)	$337,50
Cliente con rutina semanal programada.	$225	$337,50	$468 ($m^2$)	$337,50
Cliente mensual.	$225	$337,50	$468 ($m^2$)	$337,50

Comisión por Servicio

Tipo de Servicio	Comisión
Servicio de Limpieza Domestica por Hora	$ 100,00
Servicio de Limpieza de Alfombras por m^2	$ 370,00
Servicio de Limpieza de Ventanales y cortinados por Hora	$ 275,00

6.3 Estrategia de Promoción

6.3.1 Mix de Promoción

Mercado Meta:

- Clientes con ubicación geográfica de zona oeste y zona norte. Centros urbanos de Moreno, La Reja, Fco. Álvarez y Gral. Rodríguez y Pilar entre otros. Apuntando a aquellos que viven en barrios cerrados y countries.
- Familias con ingresos correspondientes a clase media / alta y clases altas. Jóvenes adultos, personas de mediana edad, adultos mayores. Personas profesionales, que valoran su tiempo libre y el cuidado del hogar.

6.3.1.1 Publicidad

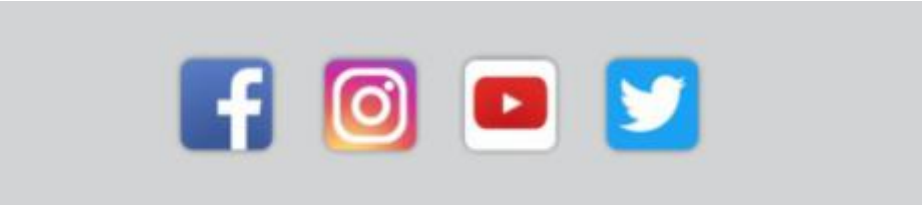
6.3.1.1.1 On-Line

Web

A través de un sitio web propio, Helper, expandir nuestro público y conquistar a nuevos consumidores en un alto porcentaje. Mediante este nuevo espacio de contacto se crea un medio de comunicación para consultas, inquietudes, sugerencias, medios de pagos, solicitud de presupuestos y contratación de servicio.

Redes Sociales

Estar presente en redes sociales como una estrategia de marketing. Las redes sociales funcionan de impulso a la hora de acercarse a los clientes, a través de notas de interés, publicidad, oferta de servicio. Como un medio alternativo de contacto con los clientes. Las que implementamos son:



- Facebook Creando un perfil, del tipo empresa, se puede subir comentarios y/o imágenes promocionando nuestros servicios y precios diariamente. De esta forma, nuestros “amigos” reciben en forma instantánea nuestras publicaciones. Además, Facebook ofrece la posibilidad de crear anuncios pagos, que es según la cantidad de clics por los interesados, que se muestran a en la pantalla principal. En el caso de publicar en nuestro “muro” el costo es cero.
- Twitter, para una rápida respuesta en 140 caracteres. Como en el caso de Facebook, al tener un perfil dentro de Twitter, se puede publicar contenido promocionando la marca con un costo nulo. También, posee avisos pagos que aparecen en el muro principal de los usuarios.
- Instagram, este medio de contacto es visual, donde se utilizará de medio de información de servicios.
- YouTube, a través de este canal, le enseñaremos como se trabaja en los diferentes servicios a través de videos demostrativos.

Mailyng

Través de newsletter a clientes se les informara de:

- Promociones.
- Nuevos servicios.
- Novedades de la empresa.
- Saludos formales en época de festividades.

6.3.1.1.2 Off Line

El contacto off line más tradicional, sigue estando presente:

- Publicidad en Suplementos de Diario Nacionales – *Clarín Zonal* – una vez al mes.
- Se harán publicaciones en revistas especializadas se dentro de los suplementos.

- Televisión: En este caso, se harán pequeños comerciales anunciando nuestros servicios en distintos horarios para captar futuros clientes y colaboradores. Se realizarán durante la noche, donde el caudal de audiencia es mayor. Se trasmitirán por *Moreno Noticias* un canal local.

6.3.1.2 Promociones de ventas

Contamos con diferentes líneas de promociones:

- *Combos*: para impulsar la contratación de más de un servicio, ofrecemos a los clientes “Combos” donde pueden obtener más de un servicio a un precio inferior a la contratación de ambos por separados.
- *Promociones mensuales*: el primer lunes de cada mes se informarán las promociones vigentes durante los primeros quince días de una vez informada. Pueden ser pack de horas a menor costo, Por ejemplo, “Mes de las cortinas “donde sale un precio atractivo para atraer clientes.
- *Posventa*: esta estrategia va a ser utilizada en aquellos clientes que hace un tiempo que no consumen nuestros servicios. Donde el impacto que debemos aplicar es mayor ya que por algún motivo se han alejado de nuestra empresa.
- *Merchandising*: ya sea en época de fin de año como con almanaques, lapiceras y llaveros.

Los canales de comunicación serán:

- On Line
- Off Line
- Telefónica.
- Personalmente en local.

6.3.1.3 ventas personales

Ventas Personales

Las ventas personales son todas las ventas en la que se establecemos directamente con el cliente, que para nosotros son fundamentales y cotidianas. Implementamos todos los medios de contactos posibles.

Atención Personal

Para tener ventas personales la empresa cuenta con personal capacitado y eficientes que posean un gran entendimiento de nuestros servicios y la calidad de ellos, los beneficios que otorgaran y poder transmitírselo a los clientes. Responder las consultas y dudas sobre nuestros servicios, además se hace un fundamental hincapié para que nuestro personal se maneje con respeto y educación hacia cada uno de nuestros clientes, tanto en el trato personal, a través de medios electrónicos como telefónico.

E-commerce

Un punto de venta personal es a través del sitio web donde el cliente puede acceder a todos nuestros servicios, lista de precios y adquirirlos. Reservar su servicio indicando día, horario y cantidad de horas.

6.3.1.4 Relaciones Publicas

Helper trabaja sus relaciones públicas a través de eventos especializados sobre limpieza doméstica y allegados. Donde podemos afianzar nuestra posición en el mercado y contactar potenciales proveedores de maquinarias para ofrecer nuevos servicios y clientes potenciales.

Por otro lado, una manera de ser reconocida es a través de ser patrocinadores de equipos de clubes de futbol infantil, entre 5 y 10 años. Donde son niños que son trasladados por sus padres y nos da el acercamiento para ampliar nuestra cartera de cliente.

6.3.2 A.I.D. A

El acrónimo AIDA significa: *atención*, *interés*, *deseo* y *acción*.

Atención: el anunciante primero debe atraer la atención del mercado meta. Una empresa no puede vender algo si el mercado no sabe que ese producto o servicio existe.

Interés: la simple conciencia de una marca rara vez lleva a una venta. El siguiente paso es crear interés en el servicio. Un anuncio impreso o un comercial de televisión no pueden señalar a los clientes potenciales todas las características y beneficios de nuestros servicios.

Deseo: hay que hacerles a nuestros compradores crear una preferencia de nuestra marca por encima de otras explicando las ventajas diferenciales del servicio por encima de la competencia.

Acción: este paso tiene que ver con crear el impulso de nuestros clientes a la compra de los servicios a través de los anuncios para comunicar de forma más eficaz las características y beneficios, y también utilizar promociones con tarjetas de créditos, diferentes medios de pago y descuentos.

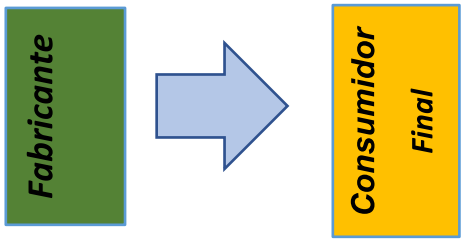
En esta sección explicaremos para aplicamos esta técnica a través de mensajes de marketing.

Helper	Atención	Interés	Deseo	Acción
Publicidad	Mailyng, radios, Clarín Zonal, Moreno Noticias. Merchandising	Tutoriales en YouTube. Fan page en Facebook y Instagram	Publicidad on-line y medios de comunicación	
Promoción de venta				Combos, Facilidades de pago.
Ventas personales	Atención personalizada			Ventas E-Commerce y Punto de venta
Relaciones públicas		Ferias Especializadas y sponsor		

6.4 Estrategia de Distribución

6.4.1 Canales de Distribución

Helper es el fabricante de sus servicios, es un canal directo con los clientes. No tercerizamos ni somos tercerizados. A través de los diferentes canales de ventas ya sean a través del sitio web, personalmente en nuestro local o telefónicamente, los clientes contactan con nosotros y pueden adquirir los servicios.



6.4.3 Funciones del Canal

E-commerce – 4to Canal

Cuarto canal, la experiencia de la compra, ya sea desde la plataforma para el escritorio o a través de la aplicación para celular.

- A través del sitio e-commerce se puede gestionar la solicitud de servicio e imprimir el comprobante con un código de solicitud de servicio o ser enviado a una casilla de correo electrónico.
- Para la comercialización de los servicios vamos a utilizar un módulo de nuestra página y el servicio de pago se hará a través de “mercado pago”.

Toda esta información que será asignada al cliente conformará el historial del cliente, el cual nos será de suma utilidad, para estadísticas y para gestiones de postventa y promociones.

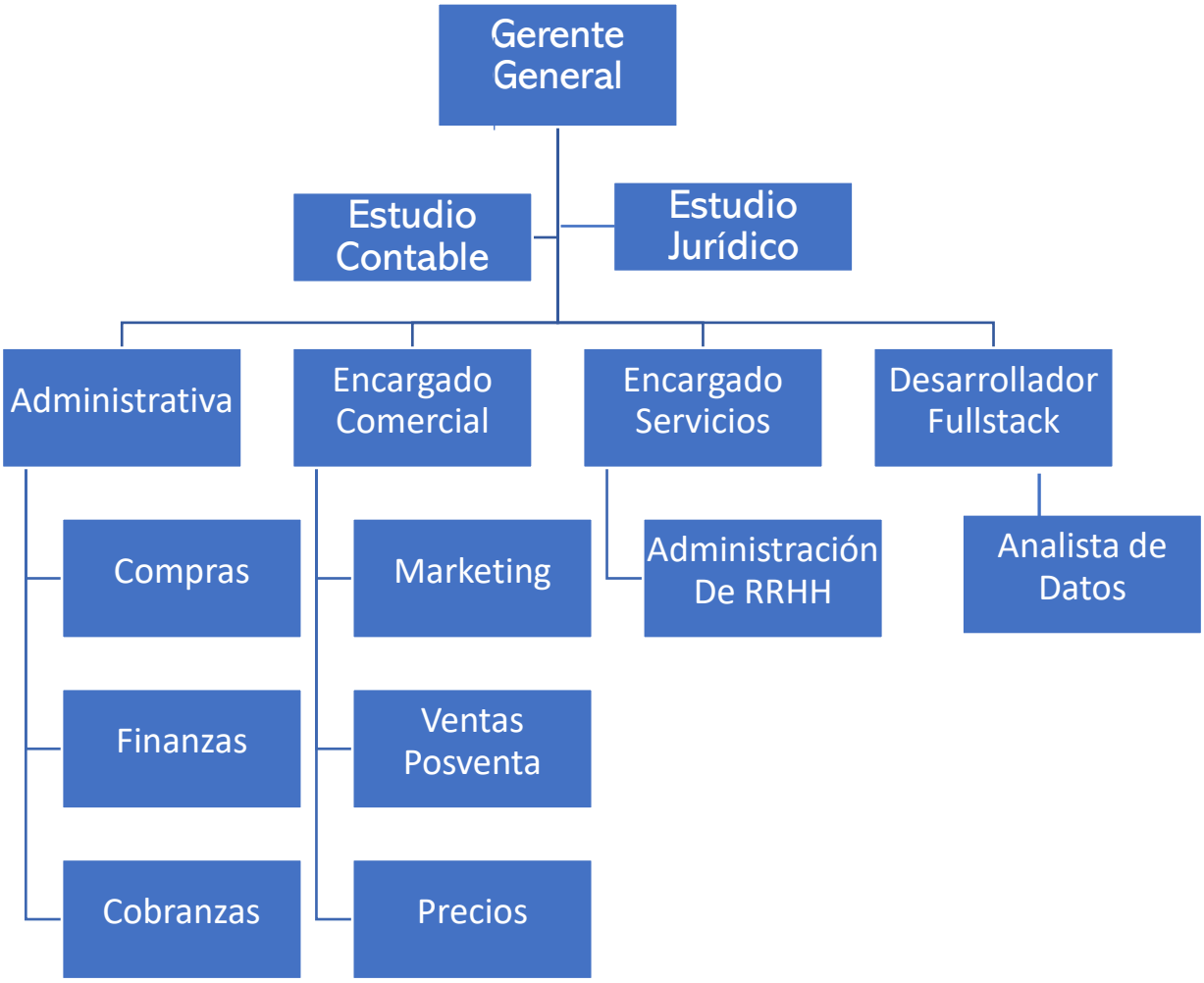
6.4.3 Logística

Los viáticos de los empleados de limpieza domestica corren a cargo de los clientes. En el caso de los servicios especiales donde incluyen maquinarias contamos con un móvil donde los empleados son trasladados junto con las herramientas de trabajo.

7. Organización Requerida

7.1 Estructura Organizacional

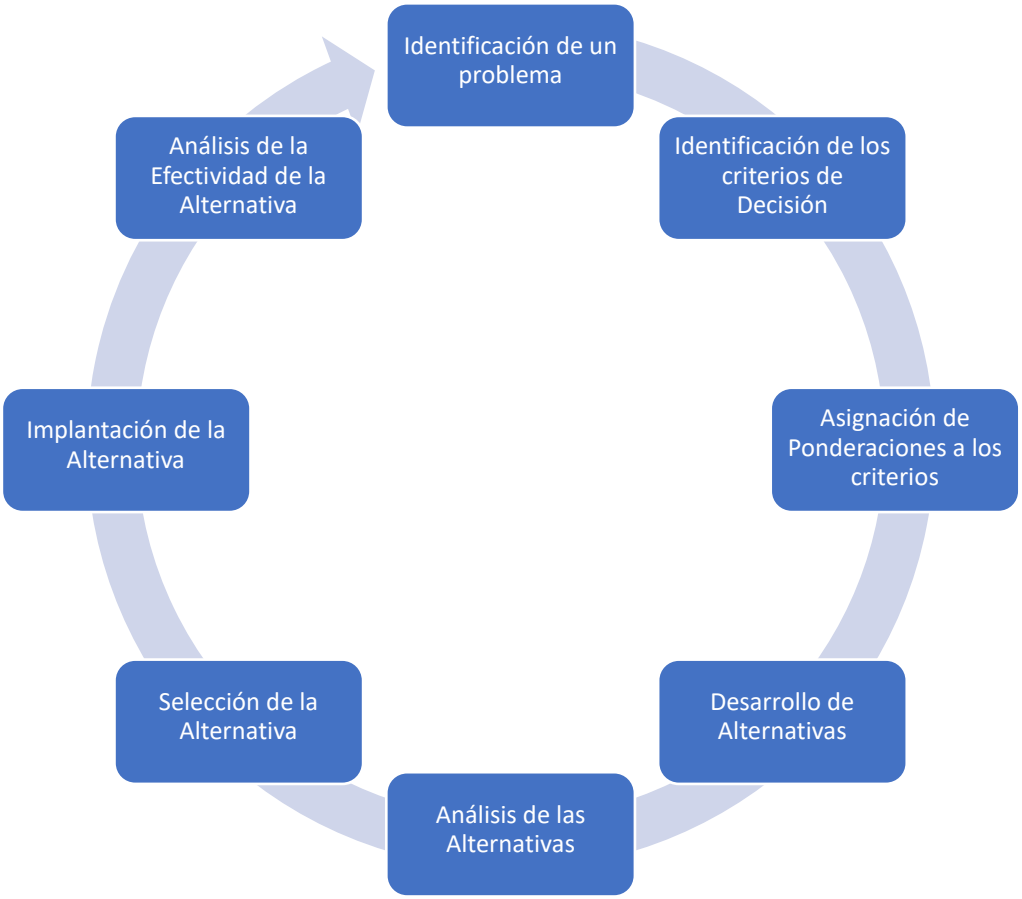
La estructura es funcional y simple. El organigrama de la empresa es de forma vertical ya que el mando se centraliza en una sola persona.



7.2 Proceso de Toma de Decisiones

La toma de decisiones en una organización se define oficialmente como el proceso para identificar y solucionar problemas. El proceso tiene dos elementos principales, el primero, identificar el problema, recabar información para medir las desventajas que provoca y la etapa del proceso de la búsqueda de la solución del problema.

El proceso para la toma de decisiones está conformado por las siguientes fases:



1. La identificación de un problema: el primer paso del proceso de toma de decisiones es haber detectado que hay una diferencia entre el estado actual de la situación y el estado deseado.

2. La identificación de los criterios para la toma de decisiones: señalar los métodos que resultarán relevantes para solucionar el problema. Cada individuo responsable de tomar decisiones en la empresa posee un abanico de criterios que lo guían en su cometido, y es importante conocer cuáles se tienen en cuenta y cuáles se omiten, ya que estos últimos resultarán irrelevantes para el encargado de tomar la decisión.

3. La asignación de ponderaciones a los criterios: priorizar de forma correcta los criterios seleccionados en la etapa anterior, puesto que no todos van a tener la misma relevancia en la toma de la decisión final

4. El desarrollo de alternativas: consiste en ser capaz de obtener y presentar todas las alternativas factibles que podrían resolver el problema con éxito.

5. El análisis de las alternativas: el responsable de la toma de decisiones en la empresa debe estudiar minuciosamente las alternativas que se han propuesto. Las fortalezas y las debilidades de cada una deberían ponerse de manifiesto de forma clara una vez comparadas con los criterios seleccionados y ordenados en la segunda y la tercera etapa. Aunque es cierto que algunas evaluaciones pueden acercarse a la objetividad, hay que tener claro que la mayoría de ellas son subjetivas a causa de su carácter de juicio de valor.

6. La selección de una alternativa: una vez establecidas y presentadas todas las alternativas, y una vez evaluadas por el responsable de la toma de decisiones según los criterios establecidos y jerarquizados, es el momento de elegir una sola alternativa: la mejor de las presentadas según el procedimiento establecido.

7. La implantación de la alternativa: Lo primero que hay que hacer para llevar a cabo la decisión es comunicarla a las personas afectadas y conseguir que se comprometan con ella. Estas decisiones se llevan a cabo desde una planificación, organización y dirección efectivas.

8. La evaluación de la efectividad de la decisión: por último, hay que evaluar el resultado conseguido a raíz de la decisión tomada y la solución adoptada y comprobar si se ha corregido el problema. Si éste todavía persiste, tendrá que estudiarse cuál de las fases anteriores resultó errónea y afrontar una nueva decisión respecto a la decisión inicial: desestimarla por completo o retomarla de forma distinta desde alguno de los pasos anteriores.

7.3 Proceso de tratamientos de conflictos

El conflicto es parte de la experiencia humana, ya sea a nivel de su grupo social o en la sociedad. Desde el núcleo familiar ya se experimentan los primeros conflictos y suelen ser los padres ò los adultos los que imponen las medidas para la conciliación para el desacuerdo entre las partes. Cuando nos insertamos en el mundo laborar ahí las medidas para manejarlos y solucionarlos es más compleja.

Existen varios métodos para llevar adelante la resolución de un conflicto a nivel organizacional. A nivel organizacional un conflicto, un desacuerdo entre las partes ya sea por recursos, por modos de procedimientos, un cruce por intereses, por ejemplo. Se debe tener en cuenta la gravedad, el impacto económico, el tiempo que lleva y actuar para lograr la conciliación.

Fisher, uno de los especialistas más conocidos en el tema de negociaciones dice: "…el conflicto de intereses en una industria en crecimiento. Todas las personas quieren participar en decisiones sobre problemas que les afectan; pero cada vez menos personas están dispuestas a aceptar decisiones dictadas por otros…".

Los Siete Elementos de Negociación de Harvard (1980) es el libro publicado por Fisher y Ury utilizaremos este método para la resolución de los conflictos.

1. Alternativas al no acuerdo

No todas las negociaciones terminan con un acuerdo. O bien se llega al acuerdo precipitados a causa de la presión que ejerce la otra parte: “Tómalo o déjalo”. Debemos tener bien preparado lo que suceda si “lo dejo”. ¿Cuáles son mis alternativas y las de ellos?

2. Intereses y posiciones

El Interés es todo aquello que no podemos intercambiar con las otras partes. Y las posiciones es la actitud que tienes acerca del tema a negociar. Un ejemplo sería: en la industria de la construcción, cuando queremos comprar un material, pero estamos en la tira y el afloje en el precio, en eso nos ofrecen uno a un menor precio, pero también a menor calidad, así que el dilema es escoger entre unos de los dos materiales Es fácil confundir los objetivos a los temas de la negociación con los intereses.

3. Opciones

Identificar toda la gama de posibilidades en que las partes pudieran llegar a un acuerdo y satisfacer sus intereses. Un acuerdo es mejor si se incorpora la mejor de muchas opciones. Puntos básicos para generar opciones:

- Conocer muy bien los procesos de crear valor, reclamar valor y la elección de REC. CREAR.
- Separar el proceso de inventar y generar ideas del de decisión. Realizar sesiones de «lluvia de ideas» con los equipos de negociación de ambas partes.
- Los mejores valores están en las diferencias. Por ejemplo, actitud frente al riesgo, cadencia temporal, percepciones, valor marginal de la misma cosa.

4. Criterios

Los criterios tienen que delinearse perfectamente para que el acuerdo al que se va a llegar deba ser con prudencia y justo para ambas partes. Es importante para nosotros que la otra parte tenga la sensación de habernos «ganado», aunque ello no sea cierto y nosotros hayamos conseguido nuestros objetivos. Este ejercicio ayudará además a descubrir nuevos intereses y opciones útiles a la negociación.

5. Relación

Las negociaciones más importantes se hacen con las personas o instituciones con las cuales hemos negociado antes y negociaremos de nuevo. Para una buena relación no debes mezclar problemas de relación personal con problemas esenciales de la negociación.

6. Comunicación

La forma que se utiliza para comunicar tiene un impacto crítico en el resultado de la negociación, especialmente cuando las dos partes no se conocen «el beneficio puede ser mucho mayor para ambos cuando se negocia cara a cara porque sólo de esta forma es posible compartir información vital.
Si el volumen de las transacciones a realizar es importante, vale la pena el ir a conocer personalmente a la persona con la que luego deberemos negociar por teléfono o e-mail. Hemos comprobado que, si ambas partes tienen en común un conocimiento personal positivo, entonces el método de comunicación (teléfono, e-mail, etc.) debe tener importancia para el buen resultado. Si los negociadores, al contrario, no se conocen o son extraños el medio se convierte en crítico y marca la diferencia.
La confianza y la aportación se establecen mucho más fácilmente cuando las partes se conocen, lo que sería imposible hacer por teléfono o e-mail entre extraños.

7. Compromisos

Los compromisos son planteamientos verbales o escritos que especifican lo que una parte hará o no hará. Podríamos llamarlos el «borrador del contrato». Pueden hacerse en el curso de una negociación. En general, un acuerdo será mejor en la medida en que las promesas hayan tenido: planteamiento, estructuración y hayan sido diseñadas para que se conviertan en acuerdos duraderos, de fácil comprensión y verificables.

7.4 Análisis de Puesto

7.4.1 Gerente General

Gerente General	
Descripción	Administrar los recursos humanos y financieros relacionados con el proceso de negocios, diseñando las políticas y procedimientos a seguir en materia presupuestal, flujos de efectivo, análisis financiero, movilidad del personal y control del gasto administrativo; buscar nuevas alternativas de servicios; garantizando el suministro oportuno de los recursos necesariamente requeridos en la implementación de los servicios.
Tareas	• Formulación y proposición de las normas, políticas y procedimientos para las actividades administrativas y la gestión contable. • Planificación de las actividades de las áreas dependientes de Administración manteniendo un control sobre el cumplimiento en tiempo y calidad de estas. • Gestión y coordinación de los recursos humanos y herramientas. • Definición de metas a corto y largo plazo y los hitos. • Elaboración, ejecución y coordinación del plan anual del presupuesto de Administración
Perfil	• **Profesión:** Administración de Empresas, Licenciado en Economía, Informática, Contaduría o afines • **Especialización:** Preferentemente en MBA o Finanzas • **Edad:** A partir de 30 años • **Idioma:** inglés avanzado • **Sexo:** Indistinto
Remuneración del mercado	• $175.500,00-

7.4.2 Encargada Administrativa

Encargada Administrativa	
Descripción	Tareas realizadas por el Gerente. Coordinar, supervisar y controlar la ejecución de los procedimientos administrativos y contables que rigen el funcionamiento interno de la organización.
Tareas	• Revisar los comprobantes de cheques, tarjetas de crédito y débito verificando que los datos estén correctos, registrados en la cuenta contable correspondiente y que tengan el soporte respectivo. • Ejecutar cierre contable mensual y anual de la organización. • Custodiar los documentos legales y financieros de la organización. • Supervisar y verificar el pago a organismos oficiales y servicios básicos. • Elaborar consolidación de gastos mensual y anual. • Tramitar todo lo concerniente a la apertura de cuentas bancarias. Controlar las obligaciones bancarias de la organización • Elaborar el flujo de efectivo y presupuesto de la organización. • Transcribir la información para la evaluación de los Estados Financieros • Revisar y controlar los archivos administrativos. • Revisar la declaración de Impuesto al Valor Agregado (I.V.A.), y realizar la declaración estimada. • Efectuar declaración de Ingresos Brutos a los diferentes Concejos Municipales. • Supervisar el cierre de auxiliares contables. Evaluar continuamente el desenvolvimiento del personal bajo su cargo. Supervisar la ejecución de los indicadores de calidad asociados al área administrativa
Perfil	• **Profesión:** licenciado contable publico • **Experiencia laboral:** 2 a 4 años en cargos similares. • **Edad:** A partir de 30 años • **Idioma:** inglés avanzado • **Sexo:** Indistinto
Remuneración del mercado	• $175.500,00-

7.4.3 Asistente Administrativo Contable

Asistente Administrativo Contable	
Descripción	Asistir al Encargado Administrativo, trabaja bajo la supervisión de este. por eso su labor suele incluir un sinnúmero de tareas destinadas a garantizar el buen funcionamiento del sector administrativo de la empresa.
Tareas	• Teneduría de libros contables: el asistente contable se encarga de llevar los libros de contabilidad de la empresa. Dicha tarea se realiza mediante la utilización de softwares exclusivos de contabilidad. • En ciertas ocasiones el contador le solicita a su asistente que prepare las facturas, que realice tareas de cuentas por pagar, y que se ponga en contacto con proveedores de servicios. • Mantenimiento de registros: el asistente contable se encarga del mantenimiento de registros financieros actuales y archivados. • Presentación de informes: el asistente contable se encarga de preparar y presentar informes. Para cumplir con dicha función, el asistente contable debe saber utilizar los softwares de contabilidad y los programas de hojas de cálculo como Microsoft Excel. Por lo general, le exigen al asistente documentos e informes con información específica como ingresos brutos, débitos mensuales y saldos de cuentas. • Cálculo de salarios: el asistente contable se encarga de calcular los salarios o las horas del empleado. • El asistente contable también es responsable de mantener el registro de los pagos efectuados a los empleados y realizar un seguimiento de los gastos del negocio.
Perfil	• **Profesión:** licenciado contable publico • **Experiencia laboral:** 1 años en cargos similares. • **Edad:** A partir de 20 años • **Idioma:** inglés básico • **Sexo:** Indistinto
Remuneración del mercado	• $58.500,00

7.4.4 Jefe de Cobranzas

Jefe de Cobranza	
Descripción	Tarea realizada por el personal Administrativo. Asistir al Encargado Administrativo, trabaja bajo la supervisión de este. por eso su labor suele incluir un sinnúmero de tareas destinadas a garantizar el buen funcionamiento del sector administrativo de la empresa.
Tareas	• Revisa comprobantes de tarjetas de crédito y débito, verificando que los datos entes correctos y registrado en la cuenta del cliente. • Emitir recibos de pagos. • implementar y controlar la estrategia de Cobranzas coordinándolo con los responsables de mora Temprana, Mora Tardía
Perfil	• **Profesión:** estudiante de carreras de administración y afines. • **Experiencia laboral:** 1 años en cargos similares. • **Edad:** A partir de 20 años • **Idioma:** inglés básico • **Sexo:** Indistinto
Remuneración del mercado	• $58.500,00-

7.4.5 Encargado Comercial

Encargado Comercial	
Descripción	Tareas realizadas por el Gerente. Planificar, organizar, dirigir, controlar y coordinar eficientemente el sistema comercial, diseñando estrategias que permitan el logro de los objetivos empresariales, dirigiendo el desarrollo de las actividades de marketing y las condiciones de venta de los productos y distribución.
Tareas	• Definir y proponer estrategias de marketing y ventas. • Definir, proponer, coordinar y ejecutar las políticas de comercialización orientadas al logro de una mayor y mejor posición en el mercado. • Evaluar la creación de nuevos productos identificando nuevas oportunidades de negocio. • Realizar las funciones de vocero autorizado ante los medios de comunicación, con la autorización de la Gerencia General, generando así corrientes de opinión favorables al desarrollo de la Empresa. • Autorizar descuentos. • Resolución de conflictos por quejas, inconvenientes y atrasos.
Perfil	• **Profesión:** Licenciado carreras de administración y afines. • **Experiencia laboral:** al menos 2 años en cargos similares. • **Edad:** A partir de 30 años • **Idioma:** inglés • **Sexo:** Indistinto
Remuneración del mercado	• $175.500,00-

7.4.6 Vendedor

Vendedor	
Descripción	Realiza tareas de ventas de servicios, atiende al público ya sea vía electrónica, telefónicamente o personal en el local. Elabora listados de servicios solicitados. Reporta ventas diarias. Personal asistente Administrativo.
Tareas	• Conocer beneficios y características del servicio. Conocimiento de las condiciones de venta. • Conocer las características de los servicios. Lista de precios y sus modificaciones. • Atender los clientes. Explicar promociones y ofertas disponibles para la venta. • Proporcionar un trato educado y cercano. • Resolver dudas sobre el servicio. • Saber usar terminales y sistemas de pedidos. • Dar servicio postventa.
Perfil	• **Profesión:** Terciario, administración de ventas. • **Experiencia laboral:** 1 año en cargos similares. • **Edad:** A partir de 20 años • **Idioma:** inglés básico • **Sexo:** Indistinto
Remuneración del mercado	• $40.959,00 -

7.4.7 Encargado de Servicios

Encargado de Servicios	
Descripción	El encargado de servicios **(**Tareas realizadas por el Gerente.**)** debe asegurarse de que todos los servicios sean eficientes y de la mejor calidad posible. Administrar el personal y servicios tomado para el cumplimiento de estos.
Tareas	• Organizar agenda diaria, asignando personal a servicios tomados. • Atender solicitudes de presupuestos. • Lograr que el personal este capacitado y motivado para cumplir los objetivos y metas, para cumplir de manera eficiente a los clientes. • Transmitir confianza a los clientes y responder las inquietudes. • Mantenimiento de las herramientas de trabajo. • Estudio y conocimiento de la competencia. • Realizar reuniones semanales para el seguimiento y apoyo de su equipo. • Revisión y autorización de los pedidos generados por los vendedores. • Confección de informes de análisis de ventas. • Establece las normas y procedimientos de atención al cliente. • Control de horas trabajadas. • Satisfacción del cliente.
Perfil	• **Profesión:** Administración de ventas, Marketing. • **Experiencia laboral:** 3 años en cargos similares. • **Edad:** A partir de 25 años • **Idioma:** inglés básico • **Sexo:** Indistinto
Remuneración del mercado	• $ 175.500,00 -

7.4.8 Personal de Servicio

Personal de Servicio	
Descripción	Personal tercerizado encargado de desarrollar las tareas designadas por el Encargado de Servicio. Atención al cliente.
Tareas	• Ejecutar tareas de aseo, lavado, planchado. • Ejecutar tareas especiales como limpieza de alfombras, cortinas y ventanales. • Buena predisposición, cortesía.
Perfil	• **Profesión:** secundario completo. • **Experiencia laboral:** 1 año en cargos similares. • **Edad:** A partir de 20 años • **Idioma:** inglés básico • **Sexo:** Indistinto
Remuneración del mercado	• Valor por Hora según la categoría.

7.4.9 Encargado de Tecnología - Desarrollador Full stack

Desarrollador Full stack	
Descripción	Responsable de funcionamiento del sitio web, software con el que se lleva adelante las tareas cotidianas, hardware y telecomunicaciones.
Tareas	• Encargado de actualización de sitio web. • Mantenimiento de los equipos ya sean terminales, impresoras, Reuters, etc. • Mantenimiento del software del negocio.
Perfil	• **Profesión:** estudiante de analista de sistemas, ing de sistemas y afines. • **Experiencia laboral:** 1 año en cargos similares. • **Edad:** A partir de 25 años • **Idioma:** inglés • **Sexo:** Indistinto
Remuneración del mercado	$84.410,00

7.4.10 Encargado Analista de Datos

Encargado Analista de Datos	
Descripción	Responsable de funcionamiento de llevar adelante el software con el que se lleva adelante la persistencia de los datos y la gestión de backup y restauraciones de datos. telecomunicaciones.
Tareas	• Encargado de actualización procedimiento de almacenamiento. • Mantenimiento de los equipos ya sean terminales, impresoras, Reuters, etc, en los cuales están involucrado la persistencia de la información.
Perfil	• **Profesión:** estudiante de analista de sistemas, ing de sistemas y afines. • **Experiencia laboral:** 1 año en cargos similares. • **Edad:** A partir de 25 años • **Idioma:** inglés • **Sexo:** Indistinto
Remuneración del mercado	$69.690,00

8. Plan Financiero

8.1 Hipótesis

Helper ofrece un servicio online de servicio doméstico. Conformado por un equipo de trabajo competente que actúa hace poco más de dos años en Zona del Partido de Moreno, La Reja, Francisco Álvarez, General Rodríguez, Pilar Provincia de Buenos Aires. Entre los servicios ofrecidos contamos con limpieza por horas, limpieza de cortinas, limpieza de alfombras, limpieza de ventanales, planes para clientes frecuentes. etc.

Cuando hablamos de hipótesis primero debemos dejar en claro nuestro mercado meta.

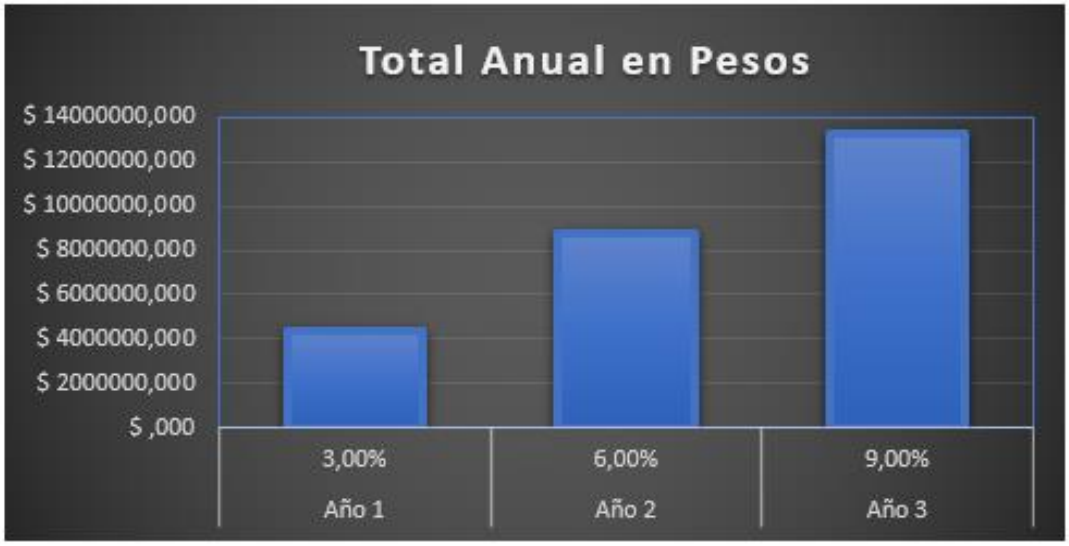
Mercado Meta	
Edad	Entre 30 y 70 años
Genero	Mujeres
Segmentacion por Ingresos	Clase medias - Clase Alta
Segmentacion por Ciclo de Vida	Personas de Mediana Edad casada o divorciada sin hijos
	Personas de Mediana Edad casada o divorciada con hijos
	Personas mayores, casadas o solteras
Lugar de Recidencia	Barrios Cerrados - Countries - Quintas

Servicios	
Comision por Tipo de Servicio	Precio
Servicio de Limpieza Domestica por Hora	$ 100,00
Servicio de Limpieza de Alfombras por m2	$ 370,00
Servicio de Limpieza de Cortinados y ventanales por Hora	$ 275,00
Cantidad de Clientes	
Según el Camara Argentina de Compras Electronicas las compras electronicas aumentaron un 76% durante el 2019 y el 90 % de los adultos argentinos conectados ya compró online alguna vez. Representa 18,3 millones de personas.	
Ingresos superiores a $80.000 a $180.000	
Ingresos del Mercado E- Commerce	
Durante el 2019 fue de $ 403.278 millones	

8.2 Modelo de ingresos

Periodo	Año 1	Año 2	Año 3
Crecimiento de Venta	3,00%	6,00%	9,00%
Total en $	$ 4.343.070,00	$ 8.787.290,00	$ 13.260.335,00
Cantidad de Horas	26.118	52.844	79746
Cantidad de m^2	1706	3452	5208

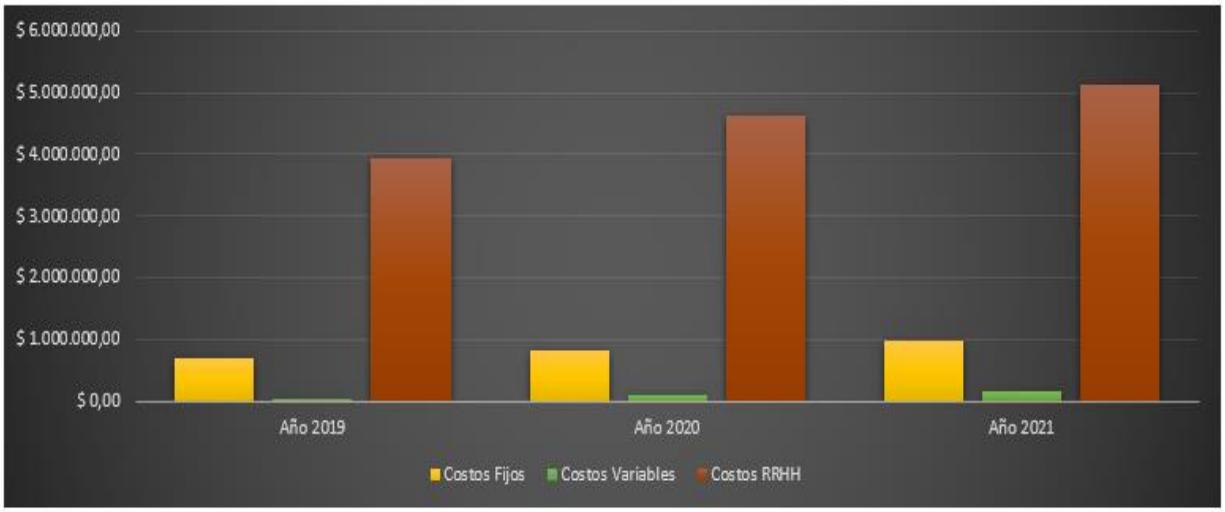
Gráfico Modelo de Ingresos



8.3 Modelo de Egresos

Periodo	Costos Fijos	Costos Variables	Costos RRHH	Total Anual
Año 2019	$ 702.966,00	$ 39.177,00	$ 3.931.200,00	$ 4.673.343,00
Año 2020	$ 833.448,00	$ 93.621,70	$ 4.633.200,00	$ 5.560.269,70
Año 2021	$ 967.380,00	$ 155.623,20	$ 5.124.600,00	$ 6.247.603,20
Total Periodo				$ 16.481.215,90

Gráfico Modelo de Egresos



8.4 Modelo de Inversión

Modelo de Inversion

Inversion	Año 0	Año 1	Año 2	Año 3
Escritorio	$ 10.197,00	$ 0,00	$ 3.399,00	$ 3.399,00
Sillas	$ 13.620,00	$ 0,00	$ 4.540,00	$ 4.540,00
Bibloteca	$ 9.567,38	$ 0,00	$ 4.783,69	$ 4.783,69
Notebook	$ 170.997,00	$ 0,00	$ 56.999,00	$ 56.999,00
Telefono Inalambrico	$ 2.998,00	$ 0,00	$ 0,00	$ 0,00
Telefono Celular	$ 6.198,00	$ 0,00	$ 0,00	$ 0,00
Desarrollo Sitio Web	$ 670.964,00	$ 0,00	$ 0,00	$ 0,00
Licencia Microsoft	$ 2.457,20	$ 0,00	$ 1.228,60	$ 1.228,60
Cafetera	$ 0,00	$ 17.999,00	$ 0,00	$ 0,00
Heladera	$ 0,00	$ 27.999,00	$ 0,00	$ 0,00
Microondas	$ 0,00	$ 11.499,00	$ 0,00	$ 0,00
Total	**$ 886.998,58**	**$ 57.497,00**	**$ 70.950,29**	**$ 70.950,29**

8.4.1 Inversión año 0

Inversion Año 0

Muebles y Utiles	Cantidad	Precio Unitario	TOTAL ($)
Escritorio	3,00	$ 3.399,00	$ 10.197,00
Sillas	6,00	$ 2.270,00	$ 13.620,00
Bibloteca	2,00	$ 4.783,69	$ 9.567,38
Hardware		**Precio Unitario**	
Notebook	3,00	$ 56.999,00	$ 170.997,00
Telefono Fijo Philips	2,00	$ 1.499,00	$ 2.998,00
Telefonos Movil Huawei ECO	2,00	$ 3.099,00	$ 6.198,00
Software	**Cantidad**	**Precio Unitario**	
Desarrollo Sitio Web	1,00	$ 670.964,00	$ 670.964,00
Microsoft Office 365 Empresa	2,00	$ 1.228,60	$ 2.457,20
TOTAL			**$ 886.998,58**

8.4.2 Inversión año 1

Inversion Año 1

Electrodomesticos	Cantidad	Precio Unitario	
Cafetera express Oster 6601R	1,00	$ 17.999,00	$ 17.999,00
Heladera 1 Frio Whirlpool Wra12a Blanco	1,00	$ 27.999,00	$ 27.999,00
Microondas BGH 20 Lt B120DB9	1,00	$ 11.499,00	$ 11.499,00
TOTAL			**$ 57.497,00**

8.4.3 Inversión año 2

Inversion Año 2

Muebles y Utiles	Cantidad	Precio Unitario	TOTAL ($)
Escritorio	1,00	$ 3.399,00	$ 3.399,00
Sillas	2,00	$ 2.270,00	$ 4.540,00
Bibloteca	1,00	$ 4.783,69	$ 4.783,69
Hardware		**Precio Unitario**	
Notebook	1,00	$ 56.999,00	$ 56.999,00
Software	**Cantidad**	**Precio Unitario**	
Microsoft Office 365 Empresa	1,00	$ 1.228,60	$ 1.228,60
TOTAL			**$ 70.950,29**

8.4.4 Inversión año 3

Inversion Año 3

Muebles y Utiles	Cantidad	Precio Unitario	TOTAL ($)
Escritorio	1,00	$ 3.399,00	$ 3.399,00
Sillas	2,00	$ 2.270,00	$ 4.540,00
Bibloteca	1,00	$ 4.783,69	$ 4.783,69
Hardware		**Precio Unitario**	
Notebook	1,00	$ 56.999,00	$ 56.999,00
Software	**Cantidad**	**Precio Unitario**	
Microsoft Office 365 Empresa	1,00	$ 1.228,60	$ 1.228,60
TOTAL			**$ 70.950,29**

8.5 Presupuesto Financiero

Presupuesto Financiero

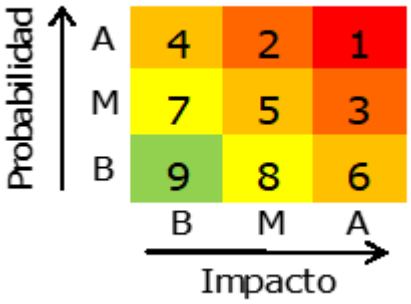
	AÑO 0	AÑO 1	AÑO 2	AÑO 3
INGRESOS	$ 0,00	$ 4.343.070,00	$ 8.787.290,00	$ 13.260.335,00
EGRESOS	$ 0,00	$ 4.673.343,00	$ 5.560.269,70	$ 6.247.603,20
Costos Fijos	$ 0,00	$ 702.966,00	$ 833.448,00	$ 967.380,00
Costos Variables	$ 0,00	$ 39.177,00	$ 93.621,70	$ 155.623,20
Costos RRHH	$ 0,00	$ 3.931.200,00	$ 4.633.200,00	$ 5.124.600,00
Utilidad antes de Impuestos	$ 0,00	-$ 330.273,00	$ 3.227.020,30	$ 7.012.731,80
IIBB 3%	$ 0,00	$ 130.292,10	$ 263.618,70	$ 397.810,05
IIGG 35%	$ 0,00	$ 0,00	$ 976.441,12	$ 2.246.013,16
INVERSIONES	$ 886.998,58	$ 57.497,00	$ 70.950,29	$ 70.950,29
Flujo de fondos	-$ 886.998,58	-$ 518.062,10	$ 1.916.010,19	$ 4.297.958,30

Tasa de corte	70%
VAN	$ 203.559,84
TIR	90%

8.6 Matriz de Riesgos

Escenario	Riesgo	Causa	Efecto	P [Ocurrencia]	Impacto	Prioridad
1	Nuevos competidores	Aislamiento social por pandemia , inhabilitacion para trabajar 7 meses	Las ventas disminuyen un 52,9% en el segundo año	Media	Media	3
2	Robos / Hurtos	Probavilidad de que el cliente sufra algun tipo de faltante.	Perdida de cliente y credebilidad, mala prensa 12%.	Baja	Alto	5
3	Competencia Desleal	Ingresa un competidor con precios desleales	Bajan las ventas 27%	Alta	Alto	3
4	Periodo de Vacaciones	Dificultad para cubrir los servicios tomados debido a contar con menos personal.	Afecta la relacion con el cliente, mayor organización de tareas en el 2do y 3er año baja 5%.	Media	Media	5
5	Disminución de las ventas en Verano	Suspensión de servicios debido a periodo vacacional de los clientes	Disminuyen las ventas un 5%	Baja	Media	8
6	Accidentes domestico	Probavilidad de que el cliente sufra rotura de algun elemento durante el servicio debido al ingreso de nuevo personal	En el 2do año aumento en los costos variables para compensar y/o reemplazar lo afectado 15%	Media	Media	5
7	Escases de RRHH	Probavilidad de no contar con suficiente personal para cumplir con todos los servicios solicitado	Disminuyen las ventas un 10%	Baja	Bajo	7
8	Competencia Desleal	Probavilidad que el cliente deje re contratar el servicio a traves nuestro	Perdida de cliente 5%.	Media	Media	5
9	Perdida de Personal Clave	Probavilidad de perder Personal clave para el funcionamiento de la empresa.		Baja	Bajo	8

10	Aumento de Costo RRHH	Al 2do año con el aumento de de los costos de recursos claves.	Aumento de costo RRHH en un 25% mas cobertura medica	Media	Media	5
11	Disminución de ingresos	Disminución del % de comision recibida para sostener precios	Perdida del 7% en comisiones recibidas.	Baja	Media	8
12	Contratacion de RRHH	Al 3er año con el aumento de actividad debemos contratar personal auxiliar.	Aumento de costo RRHH en un 9 %	Baja	Bajo	9
13	Problemas con la plataforma	Problemas con el funcionamiento del sitio E-commerce	Perdida de ingresos 5%	Baja	Media	8
14	Baja Recepcion de la plataforma	Los clientes no se familiarizan con la plataforma durante el 1er año	Bajan las ventas 10%	Baja	Medio	8
15	Error de estimacion de costos fijos	Se produce un error de calculo inicial de costos	Aumentan los costos fijos un 15%	Bajo	Medio	8



8.7 Escenarios

Escenario	Riesgo	Causa	Efecto	P [Ocurrencia]	Impacto	Prioridad
1	Nuevos competidores	Aislamiento social por pandemia , inhabilitacion para trabajar 7 meses	Las ventas disminuyen un 52,9% en el segundo año	Media	Media	3
2	Robos / Hurtos	Probavilidad de que el cliente sufra algun tipo de faltante.	Perdida de cliente y credebilidad, mala prensa 12%.	Baja	Alto	5
3	Competencia Desleal	Ingresa un competidor con precios desleales	Bajan las ventas 27%	Alta	Alto	3

Escenario 1: Las ventas disminuyen un 52,90% en el 2do año

	AÑO 0	AÑO 1	AÑO 2	AÑO 3
INGRESOS	**$ 0,00**	**$ 4.343.070,00**	**$ 4.595.752,67**	**$ 13.260.335,00**
EGRESOS	**$ 0,00**	**$ 4.673.343,00**	**$ 5.510.743,82**	**$ 6.247.603,20**
Costos Fijos	$ 0,00	$ 702.966,00	$ 833.448,00	$ 967.380,00
Costos Variables	$ 0,00	$ 39.177,00	$ 44.095,82	$ 155.623,20
Costos RRHH	$ 0,00	$ 3.931.200,00	$ 4.633.200,00	$ 5.124.600,00
Utilidad antes de Impuestos	**$ 0,00**	**-$ 330.273,00**	**-$ 914.991,15**	**$ 7.012.731,80**
IIBB 3%	$ 0,00	$ 130.292,10	$ 137.872,58	$ 397.810,05
IIGG 35%	$ 0,00	$ 0,00	$ 0,00	$ 2.246.013,16
INVERSIONES	$ 886.998,58	$ 57.497,00	$ 70.950,29	$ 70.950,29
Flujo de fondos	**-$ 886.998,58**	**-$ 518.062,10**	**-$ 1.123.814,02**	**$ 4.297.958,30**

Tasa de corte	**70%**

VAN	**$ -415.170,92**
TIR	**30%**

Escenario 2: En el 2do año perdida de cliente y credebilidad, mala prensa 12%.

	AÑO 0	AÑO 1	AÑO 2	AÑO 3
INGRESOS	**$ 0,00**	**$ 4.343.070,00**	**$ 7.732.815,20**	**$ 13.260.335,00**
EGRESOS	**$ 0,00**	**$ 4.673.343,00**	**$ 5.549.035,09**	**$ 6.247.603,20**
Costos Fijos	$ 0,00	$ 702.966,00	$ 833.448,00	$ 967.380,00
Costos Variables	$ 0,00	$ 39.177,00	$ 82.387,09	$ 155.623,20
Costos RRHH	$ 0,00	$ 3.931.200,00	$ 4.633.200,00	$ 5.124.600,00
Utilidad antes de Impuestos	**$ 0,00**	**-$ 330.273,00**	**$ 2.183.780,11**	**$ 7.012.731,80**
IIBB 3%	$ 0,00	$ 130.292,10	$ 231.984,46	$ 397.810,05
IIGG 35%	$ 0,00	$ 0,00	$ 683.128,48	$ 2.246.013,16
INVERSIONES	$ 886.998,58	$ 57.497,00	$ 70.950,29	$ 70.950,29
Flujo de fondos	**-$ 886.998,58**	**-$ 518.062,10**	**$ 1.268.667,18**	**$ 4.297.958,30**

Tasa de corte	**70%**

VAN	**$ 71.798,59**
TIR	**77%**

Escenario 3: Ingresa un competidor nuevo al mercado perdidas de ingresos en un 27%

	AÑO 0	AÑO 1	AÑO 2	AÑO 3
INGRESOS	**$ 0,00**	**$ 4.343.070,00**	**$ 8.787.290,00**	**$ 9.680.044,55**
EGRESOS	**$ 0,00**	**$ 4.673.343,00**	**$ 5.560.269,70**	**$ 6.190.022,69**
Costos Fijos	$ 0,00	$ 702.966,00	$ 833.448,00	$ 967.380,00
Costos Variables	$ 0,00	$ 39.177,00	$ 93.621,70	$ 98.042,69
Costos RRHH	$ 0,00	$ 3.931.200,00	$ 4.633.200,00	$ 5.124.600,00
Utilidad antes de Impuestos	**$ 0,00**	**-$ 330.273,00**	**$ 3.227.020,30**	**$ 3.490.021,86**
IIBB 3%	$ 0,00	$ 130.292,10	$ 263.618,70	$ 397.810,05
IIGG 35%	$ 0,00	$ 0,00	$ 1.037.190,56	$ 2.246.013,16
INVERSIONES	$ 886.998,58	$ 57.497,00	$ 70.950,29	$ 70.950,29
Flujo de fondos	**-$ 886.998,58**	**-$ 518.062,10**	**$ 1.855.260,75**	**$ 775.248,36**

Tasa de corte	**70%**

VAN	**$ -230.580,55**
TIR	**38%**

8.8 Plan de Contingencia

Escenario 1: Debido a la pandemia sufrida durante el 2020, debemos plantearnos un protocolo y un plan de contingencia en caso de una nueva ola o la aparición de un nuevo impedimento que presente una situación similar.

Las restricciones para circular y el asilamiento preventivo afectarían directamente a nuestras ventas de servicios las cuales podrían disminuir ampliamente afectando a la viabilidad del proyecto.

Este escenario lo planteamos tomando como parámetro la pandemia de Covid-19 la cual mantuvo el impedimento de servicios domésticos por ocho meses. En base a esto creeríamos conveniente aplicar las siguientes medidas para mitigar los efectos:

- Al permanecer las instalaciones cerradas se podría observar un descenso en el importe de los servicios e insumos en un 10%.

- No se contratarían nuevos recursos humanos.

- Si el aislamiento se extendería más allá de los tres meses se renegociaría el contrato de alquiler y tratando de obtener al menos un descuento del 30% por el tiempo restante que perdure la situación.

- Solicitaríamos apoyo al estado para el pago de sueldo de tres empleados por un monto que equivale a dos salarios mínimos vital y móvil ($ 33750).

Sin embargo, a pesar de tomar todas estas medidas el proyecto deja de ser viable ya que no solo se ve afectado por el distanciamiento social, sino que, además, cuando comienza a flexibilizarse no se logra recuperar el nivel de servicios contratados por la crisis económica provocada por la misma.

Plan de Contingencia

	AÑO 0	AÑO 1	AÑO 2	AÑO 3
INGRESOS	**$ 0,00**	**$ 4.343.070,00**	**$ 4.595.752,67**	**$ 13.260.335,00**
EGRESOS	**$ 0,00**	**$ 4.673.343,00**	**$ 3.796.150,19**	**$ 6.247.603,20**
Costos Fijos	$ 0,00	$ 702.966,00	$ 833.448,00	$ 967.380,00
Costos Variables	$ 0,00	$ 39.177,00	$ 44.002,19	$ 155.623,20
Costos RRHH	$ 0,00	$ 3.931.200,00	$ 2.918.700,00	$ 5.124.600,00
Utilidad antes de Impuestos	**$ 0,00**	**-$ 330.273,00**	**$ 799.602,48**	**$ 7.012.731,80**
IIBB 3%	$ 0,00	$ 130.292,10	$ 137.872,58	$ 397.810,05
IIGG 35%	$ 0,00	$ 0,00	$ 0,00	$ 2.246.013,16
INVERSIONES	$ 886.998,58	$ 57.497,00	$ 70.950,29	$ 70.950,29
Flujo de fondos	**-$ 886.998,58**	**-$ 518.062,10**	**$ 590.779,61**	**$ 4.297.958,30**

Tasa de corte	**70%**

VAN	**$ -66.179,75**
TIR	**64%**

Escenario 3: En este escenario plantearíamos que pasaría si en el tercer año se entraría de un nuevo competidor que aplicaría precios desleales por debajo del costo para ganar terreno en el mercado.

Para poder recuperarnos acudimos a nuestros clientes cautivos ofreciendo beneficios extras por el mismo precio y descuentos para futuras compras. Duplicamos el esfuerzo en publicidad logrando un 20% de aumento en el consumo.

Plan de Contingencia

	AÑO 0	AÑO 1	AÑO 2	AÑO 3
INGRESOS	**$ 0,00**	**$ 4.343.070,00**	**$ 8.787.290,00**	**$ 11.616.053,46**
EGRESOS	**$ 0,00**	**$ 4.663.343,00**	**$ 5.560.269,70**	**$ 6.251.631,22**
Costos Fijos	$ 0,00	$ 702.966,00	$ 833.448,00	$ 1.009.380,00
Costos Variables	$ 0,00	$ 39.177,00	$ 93.621,70	$ 117.651,22
Costos RRHH	$ 0,00	$ 3.921.200,00	$ 4.633.200,00	$ 5.124.600,00
Utilidad antes de Impues	**$ 0,00**	**-$ 320.273,00**	**$ 3.227.020,30**	**$ 5.364.422,24**
IIBB 3%	$ 0,00	$ 130.292,10	$ 263.618,70	$ 397.810,05
IIGG 35%	$ 0,00	$ 0,00	$ 976.441,12	$ 2.246.013,16
INVERSIONES	$ 886.998,58	$ 57.497,00	$ 70.950,29	$ 70.950,29
Flujo de fondos	**-$ 886.998,58**	**-$ 508.062,10**	**$ 1.916.010,19**	**$ 2.649.648,74**

Tasa de corte	**70%**

VAN	**$ 9.667,34**
TIR	**71%**

8.8 Plan de Salida

Luego de analizar nuestros planes de contingencia donde nuestro proyecto vuelve a ser viable, no es necesario contar con un plan de salida para ellos.

9 Factibilidades

9.1 Técnica

Gracias a las nuevas tecnologías que permiten llegar a clientes a bajo costo a través del sitio e-commerce permitiendo a los clientes tener sus perfiles de compras y consumos. Si bien es necesario actualizar las máquinas, contar con personal especializado y servidores de back-up.

Además, el sitio web cuenta con interfases muy fáciles e intuitivas de utilizar lo cual permite la rápida familiarización del cliente con esta.

Por todo lo mencionado anteriormente, creemos que la organización cuenta con los recursos y conocimiento necesarios para operar, y se considera a partir de este análisis, que el proyecto es viable desde el aspecto técnico.

9.2 Comercial

Para poder evaluar la factibilidad comercial de nuestro proyecto debemos tener en cuenta la dimensión y segmento de nuestro mercado, la oferta de servicios como así también el análisis de nuestros futuros competidores.

El entendimiento de nuestro mercado meta, clase media – alta que residen en barrios cerrados, countries o quintas, nos permitirá crear un plan de marketing, publicidad y promoción de acuerdo con ellos. El conocimiento de nuestros competidores nos permitirá poder fijar un precio razonable nuestros servicios dentro del margen propuesto por ellos.

A partir del conocimiento obtenido mencionado anteriormente, donde se tiene bien identificado nuestros clientes, se ofrece un buen valor del precio de nuestros servicios y un alto nivel de profesionalismo, vemos posible cumplir con nuestra meta de posicionamiento en el mercado.

9.3 Financiera

Luego del armado del presupuesto financiero donde se detalla el flujo de fondos de nuestro proyecto para un periodo de 3 años el cual se proyecta alcanzar un objetivo de ventas del 3% para el primer año, llegar a un 6% en el segundo y al cabo del tercer año ubicarnos en un 9% de prestadores de servicio de limpieza domestica de zona oeste.

Periodo	Año 1	Año 2	Año 3
Crecimiento de Venta	3,00%	6,00%	9,00%
Total en $	$ 4.343.070,00	$ 8.787.290,00	$ 13.260.335,00
Cantidad de Horas	26.118	52.844	79746
Cantidad de m^2	1706	3452	5208

Conociendo que para un plazo fijo tradicional se obtiene casi un 32,3 % anualmente, se fijó una tasa de corte del 70% y así ser un poco más tentador en el caso se necesitar atraer futuros inversores en caso de necesitarlo. Con esta tasa elegida y armado el presupuesto se concluye que el proyecto es viable.

Además, se realizó un análisis de posibles riesgos donde los tres más significativos fueron analizados en distintos escenarios y así poder observar el posible impacto sobre nuestro negocio. Dos de ellos ponen en peligro el proyecto al no alcanzar la tasa de corte elegida por lo cual se proyectaron nuevos presupuestos modificando los valores definidos en el plan de contingencia correspondiente demostrando que, en caso de suceder lo mencionado en el riego y aplicando las acciones detalladas en el plan de contingencia, el proyecto vuelve a ser viable.

9.4 Legal

Registro de la Sociedad

Como emprendedores buscamos maneras de registrarnos legalmente al mercado, analizando diferentes alternativas y beneficios. Comparando las opciones como SAS, SRL y SA.

	RUBROS	S.A.S.	S.R.L.	S.A.
Estatuto o Contrato Constitutivo	Estatuto modelo	Sí, diseñado por IGJ.	No hay modelo	
	Naturaleza Jurídica	Contrato adhesión y de consumo del estatuo modelo	Contrato consensual	
	Estatuto adecuado al emprendimiento empresarial	Plena autonomía de la voluntad para redactar cláusulas	Debe adecuarse a los requisitos del tipo regulado en la LGS, de lo contrario queda encuadrado en la Sección IV (LGS)	
	Plazo inscripción	24 horas		
	Forma de constitución	- Instrumento público - Instrumento privado con firma certificada digitalmente. - InsEstatuto modelo con firma certificada digitalmente.	- Instrumento público - Instrumento privado con firma certificada.	- Instrumento público
Accionistas y socios	Cantidad	Una o varias personas.	2 a 50 personas.	Al menos 2 personas.
	Responsabilidad	- Por la integración de las acciones o cuotas que suscriban o adquieran, respectivamente. - Por la integración de aportes del resto de socios, en forma solidaria e ilimitada.		Limitada a la integración de las acciones que suscriban o adquieran.
	Capital social	Dividido en acciones, no podrá ser inferior inferior a 2 salarios mínimos (ahora $21.400).	2 a 50 personas.	Al menos 2 personas.

Fuente: Estudio Policella. El Cronista

Luego de este análisis consideramos que las formas de sociedad SAS (Sociedades de Acciones Simplificada) es la que mejor se adapta y nos brinda los siguientes beneficios:

1. **Simples.** Son más sencillas de constituir y más económicas respecto de las sociedades tradicionales, las Sociedades Anónimas (SA) y las Sociedades de Responsabilidad Limitada (SRL), que son las formas legales más habituales hasta ahora en la Argentina.

2. **No se necesitan socios.** A diferencia de las SRL o las SA, que requieren por lo menos dos socios para constituirse, las SAS pueden ser unipersonales, es decir, estar en manos de un único emprendedor.

3. **Sin límites de socios.** Tampoco tienen un número máximo de integrantes, como sí ocurre en las SRL, donde se acepta hasta 50 asociados.

4. **En pocos días.** Si se utiliza el modelo tipo previamente aprobado, la inscripción puede realizarse hasta en 24 horas, de acuerdo a lo dispuesto en el art. 38 de la mencionada ley 27.349 (Ley Pymes).

5. **Digitales**. Se promueve la digitalización de las inscripciones societarias, así como las notificaciones por la misma vía, lo que acelera significativamente todos los plazos de registración y modificaciones estatutarias. A la inscripción, se obtiene la Clave Única de Identificación Tributaria (CUIT) online.

6. **Fácil aumentar el capital.** También está autorizada la integración de un aumento de capital inferior al 50% del mismo sin necesidad de publicar edictos ni registración. Se faculta a realizar aportes irrevocables por 24 meses.

7. **Clases de acciones.** Pueden constituirse por instrumento público o privado y su capital puede dividirse en acciones, que a su vez admiten distintas clases (ordinarias, preferidas o escriturales), lo que no ocurre en las SRL.

8. **Facturas.** La AFIP le otorga factura tipo A automáticamente, al momento de la constitución, no ocurriendo lo mismo con otro tipo de sociedades.

9. **Administración unipersonal.** La administración puede ser unipersonal con la designación de un suplente, salvo que se haya propuesto un síndico. Su duración puede ser indeterminada y la fijará el estatuto social.

10. **Capital inicial bajo.** El capital mínimo que requiere para su constitución es muy bajo. Es el equivalente a dos salarios mínimos, vitales y móviles, a diferencia de la sociedad anónima (SA) donde se exige la suscripción mínima de $ 100.000.

11. **Separación del capital personal.** Reduce los riesgos, por ejemplo frente a una inscripción como monoributista, ya que los accionistas de una SAS limitan su responsabilidad a la integración de las acciones que suscriban o adquieran.

12. **Objeto social amplio** El objeto social podrá ser plural y deberá enunciar en forma clara y precisa las actividades principales que constituyen el mismo

13. **Control.** No necesitan sindicatura, a diferencia de la sociedad anónima unipersonal, que sí la requiere. Tampoco estarán bajo la fiscalización de la Inspección General de Justicia (IGJ), por lo que no deben presentar sus estados contables a este organismo, como sí deben hacerlo las SA.

Registro de Dominio Web

ARTICULO 2°- NIC Argentina efectuará el registro de los nombres de dominio de Internet, en adelante "nombres de dominio", bajo las zonas que determine, con los requisitos y a través de los procedimientos que al efecto establezca, con excepción de la zona denominada ".edu.ar", la que será gestionada por la ASOCIACIÓN REDES DE INTERCONEXION UNIVERSITARIA - ARIU.

ARTICULO 3°- El Usuario que registre un nombre de dominio revestirá la calidad de Titular de este, manifestando conocer y aceptar inexcusablemente las normas y procedimientos que rigen al respecto, asumiendo que el registro no se realiza de mala fe, con un propósito ilegal, ni afecta o perjudica a otros Usuarios y/o terceros.

La marca ***Helper*** está en proceso de registro y el patentamiento de sus servicios en estado "Iniciado" sin oposiciones hasta el momento.

Asesores Legales

Por otro lado, ***Helper*** solo cobra comisión por los servicios prestados por el personal tercerizado. No tenemos relación directa con los clientes. Sin embargo, contamos con asesor de legales en caso de que se presente alguna situación que lo amerite, por ejemplo, un caso de hurto o situación que lo amerite en algún Domicio.

Por todo esto, se comprueba que el proyecto es viable legalmente.

Trabajo Final de Ingeniería

Proyecto: Helper

Tema: Aspectos descriptivos de la Solución Tecnológica

Alumno: Natalia M. Gonzalez

Legajo: T1-5132

Sede: Castelar

Profesor: Parkinson, Christian

10.2 Índice

Trabajo Final de Ingeniería **127**

10.2 Índice **128**

10.3 Historial de Revisión **133**

10.4 Introducción **134**

10.4.1 Propósito 134

10.4.2 Alcance 135

10.5 Descripción de partes Interesadas (Participantes en el proyecto) y Usuarios **136**

10.5.1 Resumen de las Partes Interesadas 136

10.5.1 Entorno del Usuario 137

10.5.2 Entorno de Usuario 138

10.5.3 Perfil de las Partes Interesadas 138

10.6 Descripción Global del Servicio **141**

10.6.1 Perspectiva del Servicio 141

10.6.2 Resumen de Características 142

10.7.2 Descripción Detallada del Servicio **143**

10.7.2.1 REQ001 – Alta Cliente 143

10.7.2.3 REQ003 – Catalogo de Servicios 143

10.7.2.4 REQ004 – Venta mediante sitio E-Commerce 143

10.7.2.5 REQ005 – Solicitud de Presupuestos 143

10.7.2.6 REQ006 – Cancelación de Venta de Servicio 144

10.7.2.7 REQ007 – Alta Empleado 144

10.7.2.8 REQ008 – Asignación de Perfiles 144

10.7.2.9 REQ009 – Consultar Empleado 144

10.7.2.10 REQ010- Generar orden de trabajo 144

10.7.2.11 REQ011- Asignación orden de trabajo 144

10.7.2.12 REQ012 – Estadísticas de trabajos 144

10.7.2.13 REQ013 – Facturación 145

10.7.2.14 REQ014 – Emisión de Nota de Crédito 145

10.7.2.15 REQ015 – Emisión de Nota de Debito 145

10.7.2.16 REQ016 – Cobranza 145

10.7.2.17 REQ017 – Multidioma 145
10.7.2.18 REQ018 – Registrar en Bitácora 145
10.7.2.19 REQ019 – Log in 146
10.7.2.20 REQ020 – Log Out 146
10.7.2.21 REQ027 – Alta Servicio 146
10.7.2.22 REQ028 – Modificar Servicio 146
10.7.2.23 REQ029 – Eliminar Servicio 146
10.7.2.24 REQ030 – Modificar Empleado 146
10.7.2.25 REQ031 – Eliminar Empleado 146
10.7.2.26 REQ032 – Alta Dirección 146
10.7.2.27 REQ033 – Eliminar Dirección 147
10.7.2.28 REQ034 – Alta Tarjeta 147
10.7.2.29 REQ035 – Modificar Tarjeta 147
10.7.2.30 REQ036 – Eliminar Tarjeta 147
10.7.2.31 REQ037 – Recuperar Contraseña 147
10.7.2.32REQ043 – Genera Recibo 147
10.7.2.33REQ041 – Consulta Factura 148
10.7.3 Requisitos de Desempeño del Sistema 149
10.7.4 Requisitos de Entorno 151
10.7.4.1 REQ022 – Asignar permisos a los Roles 151
10.7.4.2 REQ023 – Generar Dígitos de Verificador Vertical 151
10.7.4.3 REQ024 – BackUps 152
10.7.4.4 REQ025 – Restore 152
10.7.4.5 REQ026 – Ver Logs 152
10.7.4.6 REQ038 – Genera Digito Verificador Horizontal 152
10.7.4.7 REQ039 – Generar Digito Verificador Vertical 152
10.7.4.8REQ43 – Modificar permisos a los Roles 153
10.7.4.9 REQ044 – Modificar Orden de trabajo 153
10.7.4.10 REQO45 – Consultar Roles 153
10.10 Caso de Uso **154**
Requisitos involucrados al sector comercial **154**
CU - 001 - Alta del Cliente 155
Diagrama de Secuencia CU - 001 – Alta del Cliente 160
CU - 002 – Reporte del Cliente 162

Diagrama de Secuencia CU- 002 – Reporte del Cliente 167
CU - 003 – Catalogo de Servicios 168
Diagrama de Secuencia CU - 003 – Catalogo de Servicios 172
CU - 004 – Venta mediante sitio E-Commerce 174
Diagrama de Secuencia CU - 004 – Venta mediante sitio E-Commerce 179
CU - 005 – Solicitud de Presupuesto 184
Diagrama de Secuencia CU - 005 – Solicitud de Presupuesto 186
CU - 006 – Cancelación de Venta 188
Diagrama de Secuencia CU - 006 – Cancelación de Venta 192
CU - 027 – Alta Servicio 194
Diagrama de Secuencia CU - 027 – Alta de Servicio 198
CU - 028 – Modificar Servicio 200
Diagrama de Secuencia CU - 028 – Modificar Servicio 204
CU - 029 – Eliminar Servicio 206
Diagrama de Secuencia CU - 029 – Eliminar Servicio 210
CU - 032 – Alta dirección 212
Diagrama de Secuencia CU - 032 – Alta dirección 216
CU - 033 – Eliminar dirección 218
Diagrama de Secuencia CU - 033 – Eliminar dirección 222
CU - 034 – Alta Tarjeta 224
Diagrama de Secuencia CU - 034 – Alta Tarjeta 228
CU - 035 – Modificar Tarjeta 230
Diagrama de Secuencia CU - 035 – Modificar Tarjeta 234
CU - 036 – Baja Tarjeta 236
Diagrama de Secuencia CU - 036 – Baja Tarjeta 240
CU - 037 – Recuperar Contraseña 242
Diagrama de Secuencia CU - 037 – Recuperar Contraseña 246
Requisitos relacionados a Personal **248**
CU - 007 – Alta Empleado 249
Diagrama de Secuencia CU - 007 – Recursos Humanos 254
CU - 008 – Asignación de Perfiles 256
Diagrama de Secuencia CU - 008 – Asignación de Perfiles 260
CU - 030 – Modificar Empleado 262
Diagrama de Secuencia CU - 030 – Modificar Empleado 267

CU - 031 – Eliminar Empleado 269
Diagrama de Secuencia CU - 031 – Eliminar Empleado 273
CU - 009 – Consultar Empleado 275
Diagrama de Secuencia CU - 009 – Consultar Empleado 279
Requisitos relacionados organización de Recursos 281
CU - 010 - Generar orden de trabajo 282
Diagrama de Secuencia CU – 010 – Realizar de Orden de Trabajo 286
CU - 011 – Asignación de Orden de Trabajo 287
Diagrama de Secuencia CU – 011 – Asignar de Orden de Trabajo 291
CU - 012 – Estadísticas de trabajos 293
Diagrama de Secuencia CU - 012 – Estadísticas de trabajos 297
CU - 044 - Modificar orden de trabajo 298
Diagrama de Secuencia CU - 044 – Modificar Orden de trabajo 302
Requisitos relacionados a Ventas 303
CU - 013 – Facturación 304
Diagrama de Secuencia CU - 013 – Facturación 307
CU - 014– Emisión Nota de Crédito 310
Diagrama de Secuencia CU - 014 – Emisión Nota de Crédito 313
CU - 015 – Emisión de Notas de Debito 316
Diagrama de Secuencia CU - 015 – Emisión de Notas de Debito 321
CU - 016 - Cobranza 322
Diagrama de Secuencia CU - 016 – Cobranza 327
CU - 044 -Genera Recibo 329
Diagrama de Secuencia CU - 044 -Genera Recibo 333
CU - 041 – Consultar Factura 334
Diagrama de Secuencia CU - 041 – Consultar Factura 338
Requisitos Especificaciones Adicionales 340
CU - 017 – Multidioma 341
Diagrama de Secuencia CU - 017 – Multidioma 346
CU - 018 – Registrar en Bitácora 347
Diagrama de Secuencia CU - 018 – Ver Logs 351
CU - 019 – Log in 352
Diagrama de Secuencia CU - 019 – Log in 357
CU - 020 – Log Out 359

Diagrama de Secuencia CU - 020 – Log Out 362
CU – 024 - Back Up 363
Diagrama de Secuencia CU - 024 – BackUp 364
CU – 025- Restore 368
Diagrama de Secuencia CU - 025 – Restore 373
CU - 026 – Ver Logs 374
Diagrama de Secuencia CU - 026 – Ver Logs 378
CU – 038 – Genera Digito Verificador Vertical 380
Diagrama de Secuencia 038 – Genera Digito Verificador Vertical 384
CU – 39 – Generar Digito Verificador Horizontal 385
Diagrama de Secuencia CU – 39 – Generar Digito Verificador Horizontal 389
CU – 43 – Modificar permisos a los Roles 390
Diagrama de Secuencia CU – 43 – Asignar permisos a los Roles 394
CU – O45 – Consultar Roles 396
Diagrama de Secuencia CU – 44 – Consultar Roles 400
Pantalla 401
10.11 DER (Diagrama Entidad Relación) **402**
10.13 Diagrama de Clases **403**
10.14 Diagrama de Contexto **404**
10.16 Mapa de Navegación **405**
10.16 Diagrama de Despliegue **406**
10.17 Diagrama de Componentes **407**
10.18 Diagrama de Paquetes **408**

10.3 Historial de Revisión

Fecha	Versión	Autor	Descripción
14/05/2018	0.0	Natalia Gonzalez	Creación de documento visión.
04/06/2018	0.1	Natalia Gonzalez	Requerimientos.
11/11/2019	0.2	Natalia Gonzalez	Caso de usos y diagramas de secuencia. Diagrama Entidad Relación y Diagrama de Clases.
03/12/2020	0.3	Natalia Gonzalez	Diseño de diagrama de caso de uso, diagrama entidad relación, diagrama de clases, diagrama de secuencia en Enterprise Architect

10.4 Introducción

10.4.1 Propósito

El propósito de este documento es mostrar el proceso que va a llevar adelante la construcción del sistema Helper, que aumentar el control sobre sus tareas cotidianas.

El sistema Helper facilita la comunicación y el acercamiento al cliente. Agilizando la contratación de servicio doméstico de manera on-line, solicitudes de presupuestos según las categorías de servicios que prefiera a través del sitio web. Ofrecemos una página amigable e intuitiva. Permite responder al cliente en un tiempo más reducido. Brindando un intercambio ágil con el mismo. Permite sacar estadísticas de los consumos de los clientes, sirviéndonos para identificar la asiduidad de sus consumos.

Helper es un sistema que permite al empleador el control de su personal. Es de fácil uso y concede llevar a cabo las tareas relacionadas con el control de la nómina. Asignación de personal a los trabajos solicitados por los clientes. Emisión de reportes de horas trabajadas por cada empleado para la liquidación de honorarios.

Por otro lado, Helper ofrece la organización de los servicios por empleado. Permitiendo llevar una agenda de trabajos programados. Posibilitando la emisión de un reporte con el detalle de servicios brindados por cada empleado y estadísticas por satisfacción del servicio. El sistema busca manejar de manera segura la información cotidiana sobre los servicios tomados, empleado a cargo de llevar a cabo dicha tarea, horas programadas y costos de este.

10.4.2 Alcance

El propósito de este documento es detallar, dar una visión del funcionamiento, las características y bondades que ofrecería la página web del sistema de personal doméstico Helper el cual se va a desarrollar. El alcance del sistema está determinado por las tareas que se pueden llevar a cabo. Funcionalmente se podrá manejar la administración de los servicios, gestión del personal y administración de las ventas.

Por parte del cliente:

- ✓ Podrá acceder a ver los servicios disponibles para adquirir.
- ✓ Registrarse en la Página.
- ✓ Tener su perfil personal y registros de sus compras.

Por parte de la Empresa:

- ✓ Registro del personal.
- ✓ asignación del personal a los servicios adquiridos por los clientes.
- ✓ asignación de Perfiles.
- ✓ Emisión de comprobantes (facturación, Notas de crédito, Notas de Débito, comprobante de pago)

Seguridad:

- ✓ Backup
- ✓ Restore
- ✓ Log in Y Log out
- ✓ Bitacora
- ✓ Digito Verificador
- ✓ Asignación de perfiles (patente - familia)

10.5 Descripción de partes Interesadas (Participantes en el proyecto) y Usuarios

Para la identificación de las tareas se ha recurrido al personal que interviene en las actividades de las cuales van a surgir las necesidades a cubrir. Luego de varias reuniones con personal de recepción y coordinación se han identificado los diferentes procesos, requerimientos y sus diferentes necesidades para lograr el modelado de estos.

A continuación, se detallará el perfil de los usuarios, participantes y todos aquellos que son necesarios para el proyecto.

10.5.1 Resumen de las Partes Interesadas

Nombre	Descripción	Responsabilidad
Carlos Báez	Gerente General	- Seguimiento de desarrollo del proyecto. - Aprobación de costos, alcances y avances.

10.5.1 Entorno del Usuario

Rol	Descripción	Nombre	Área Funcional
Recepcionista	Responsable de llevar a cabo las actividades: Atención Telefónica. Atención personalizada Cobro de servicios. Emisión de recibos de pago de servicios.	Marta Cuesta	Recepción.
Coordinador	Asistente del gerente General: Responsable de la organización de horarios. Asignación de tareas. Organización del personal.	Juan Rodríguez	Coordinación.
RRHH	Control de Ausentismo. Control de Horarios. Licencias. Certificados médicos. Personal ABMC	Laura López	RRHH
Ventas	Se ocupa de responder a presupuestos. Ordenes de trabajo. Descuentos. Promociones. Servicio Posventa.	Sabrina López	Ventas

10.5.2 Entorno de Usuario

El usuario para poder acceder al sistema debe contar con una computadora o dispositivo móvil con acceso a internet. A través de un navegador va a poder accedes a la interfaz del sistema Helper y de acuerdo con su perfil va a tener diferente menú y características.

La interfaz del sistema Helper es clara e intuitiva al transmitir la información de manera precisa para evitar que el usuario cometa errores durante la interacción ya que cuenta con controles de validación de datos.

Por otro lado, es flexible ya que sea por parte del Usuario cliente o administración permite que restaure o deshaga acciones.

A nivel estético, el sitio es atractivo y armónico, manteniendo la paleta de colores y estilos según el tipo de Usuario durante toda la interacción con el mismo.

El objetivo del sitio es que la interacción del usuario con Helper sea amigable, claro e intuitiva ya que de eso depende la venta de nuestros servicios y ganar mercado.

10.5.3 Perfil de las Partes Interesadas

Cada uno de los usuarios puede ingresar al sistema a través de su usuario y contraseña lo cual le va a permitir acceder de acuerdo con su perfil a diferentes funcionalidades. Por ejemplo: en el caso del perfil de ventas este podrá ingresar para dar de alta un nuevo servicio.

Para poder acceder al sistema, este, debe estar instalado en una terminal o servidor y acceder a él a través de la red LAN de la organización. Simplemente a través de teclado y ratón se va a poder interactuar con el sistema, siendo el mismo simple e intuitivo ya que posee el aspecto similar a cualquier aplicación en entorno de Microsoft Windows.

Representante del Departamento de Sistemas

Representante	Responsable de la Gestión del Proyecto
Descripción	Gerente General
Tipo	Usuario Clave.
Responsabilidad	Responsable general de la coordinación. Asignación de horas y responsables.
Criterio de éxito	Llevar a cabo una organización de la establecimiento coordinada y prolija.
Grado de Participación	Alto.
Comentarios	Las propuestas de cambios deben presentarse y defenderse ante él para que luego sean aprobadas.

Representante	Control de Horas
Descripción	RRHH
Tipo	RRHH
Responsabilidad	Responsable de mantener actualizado la base de datos del personal. Alta, baja, modificación, consulta.
Criterio de éxito	Ausencia de errores en la liquidación de haberes.
Grado de Participación	Moderado
Comentarios	Ninguno.

Representante	Créditos y Cobranzas
Descripción	Responsable de Cobranzas, facturación.
Tipo	Usuario Experto.
Responsabilidad	Emitir los reportes por horas para la facturación del servicio ofrecido por la empresa.
Criterio de éxito	Cobrar todos los servicios tomados. Evitar morosos y/o incobrables.
Grado de Participación	Moderado.
Comentarios	Sus opiniones son tomadas muy en cuenta.

Representante	Coordinador
Descripción	Coordinación de tareas. Asigna las tareas al personal, organización y gestión
Tipo	Usuario experto.
Responsabilidad	Organizar eficientemente los recursos.
Criterio de éxito	Clientes satisfechos.
Grado de Participación	Alta.
Comentarios	Sus opiniones son tomadas muy en cuenta.

Representante	Ventas y Marketing
Descripción	Ventas. Fijar precios, promociones, responder a presupuestos. Servicio posventa. Reportes.
Tipo	Usuario Experto.
Responsabilidad	Cartera de clientes.
Criterio de éxito	Clientes satisfechos.
Grado de Participación	Alta.
Comentarios	Sus opiniones son tomadas muy en cuenta, detecta a través de los repostes bajas en las tasas de uso, importante a la hora de otorgar descuentos.

10.6 Descripción Global del Servicio

10.6.1 Perspectiva del Servicio

El sistema por desarrollar es para la Agencia de Servicio Doméstico Helper con la intención de poder sistematizar y agilizar el alta de servicios, asignaciones de fechas y horarios, gestionar sus pagos, control de personal. El sistema permite el alta de un servicio tomado con la asignación del personal a cargo, fecha, horario, localización. Detalle y estadísticas de cada uno de los clientes y empleados.

10.6.2 Resumen de Características

Características	Beneficio para el Cliente
Mayor acercamiento a clientes y potenciales clientes a través del sitio web.	Registro de cliente. Adquisición de servicios. Abono de Servicios. Reporte de estado. Solicitud de presupuestos. Historial.
Organización de servicios tomados	Permite la organización de los trabajos y a que personal fue asignado. Emite reportes por empleado, agenda de trabajos otorgados. Modificar servicio-responsable
Mayor Control	Permite reportes por clientes, empleados, trabajos, para poder detectar falencias y virtudes para realizar las modificaciones para potenciar o desactivar según sea necesario.
Ordenes de trabajo	El cliente a través de su perfil puede optar por la cancelación del servicio y liberando los recursos para la asignación a otra tarea.
Gestor Empleado	Administración de toda la información relacionada al empleado. Registro de empleado. Modificación del empleado. Búsqueda del empleado. Horas y días trabajados. Servicios asignados.

10.7.2 Descripción Detallada del Servicio

El objetivo de Helper es llevar el control sobre la gestión de los trabajos tomados, seguimiento y supervisión. El coordinador se ocupa de la atribución de tareas al personal. Es el que asigna los servicios tomados. El personal con el perfil de recursos humanos tendrá acceso a poder dar de alta, baja y modificar los datos referentes al personal. El coordinador le indicara que servicio fue destinado para cada persona según la tarea y el personal de recursos humanos modificara la planilla de agenda.

10.7.2.1 REQ001 – Alta Cliente

Se brinda la posibilidad de crear un nuevo cliente, ingresar todos sus datos personales incluyendo usuario (correo electrónico) y contraseña en el sitio para poder acceder a comprar servicios. Carga de datos personales.

10.7.2.3 REQ003 – Catalogo de Servicios

A través del sitio e-commerce el cliente podrá acceder al catálogo de servicios disponibles para la compra.

10.7.2.4 REQ004 – Venta mediante sitio E-Commerce

Se deben poder a través del sitio e-commerce los clientes acceder a la compra de servicios disponibles. Programando fecha de servicio, cantidad de horas, tipo de servicios y registrar el pago.

10.7.2.5 REQ005 – Solicitud de Presupuestos

Se deben poder generar presupuestos a través del sitio e-commerce mediante un formulario la cotización de servicios donde se podrá visualizar nombre del servicio, precio unitario, cantidad e importe total.

10.7.2.6 REQ006 – Cancelación de Venta de Servicio

A través del sitio e-commerce el cliente podrá eliminar el servicio pedido.

10.7.2.7 REQ007 – Alta Empleado

Permite al encargado de recursos humanos llevar adelante la carga de los datos relacionados al alta del empleado.

10.7.2.8 REQ008 – Asignación de Perfiles

Se debe poder llevar un control sobre la asignación de perfiles de acuerdo con su jerarquía y sector dependerá a que módulos puede ingresar. Asignando usuario y contraseña.

10.7.2.9 REQ009 – Consultar Empleado

A través del sitio el empleado de recursos humanos podrá listar y consultar todos los datos involucrados con el empleado. Tanto sus datos personales como perfil asignado.

10.7.2.10 REQ010- Generar orden de trabajo

Al momento en que un cliente solicita un servicio se realiza una orden de trabajo a efectos de avisar a las distintas áreas intervinientes en el proceso de la venta, coordinación, asignación de recursos, poder realizar un seguimiento y asociación a través de este documento.

10.7.2.11 REQ011- Asignación orden de trabajo

Al momento en que un cliente solicita un servicio y se realiza una orden de trabajo el coordinador debe asignar a dicha orden de trabajo las personas involucradas a llevar a cabo dicho servicio contratado. Actualizando de esta manera la planilla de servicios tomados donde queda registrado los datos del cliente (nombre y apellido, dirección, tel., etc.), servicio (día, cantidad de horas, tipo de servicio) y el personal involucrado.

10.7.2.12 REQ012 – Estadísticas de trabajos

Se debe poder a través del perfil de encargado acceder a informes de trabajos realizados ya sea por tipo de tarea realizada o por personal a cargo de dicha.

10.7.2.13 REQ013 – Facturación

A través del perfil de encargado de ventas podrá emitir la factura correspondiente a los servicios comprados pendientes de facturación por el cliente.

10.7.2.14 REQ014 – Emisión de Nota de Crédito

A través del perfil de encargado de ventas podrá emitir la nota de crédito por la cancelación de los servicios comprados por el cliente.

10.7.2.15 REQ015 – Emisión de Nota de Debito

A través del perfil de encargado de ventas podrá emitir la nota de débito por algún cargo excedente de los servicios comprados por el cliente.

10.7.2.16 REQ016 – Cobranza

A través del perfil de encargado de ventas permite registrar el pago realizado por el cliente por la cancelación de los servicios comprados y podrá emitir el comprobante de pago por la cancelación de los servicios comprados por el cliente.

10.7.2.17 REQ017 – Multidioma

Permite seleccionar el idioma en el que se desea que se muestre el sitio web.

10.7.2.18 REQ018 – Registrar en Bitácora

En ella deben quedan registradas todas las operaciones que realicen los usuarios durante la utilización del sistema. Esto permitirá hacer un trazado de las actividades desarrolladas por el usuario dentro de la aplicación. Los datos que la bitácora incluye son fecha, hora, usuario, actividad, información asociada con la actividad.

Dicha bitácora se grabará en el Log de eventos de Windows y además en un archivo TXT aparte.

10.7.2.19 REQ019 – Log in

Permite verificar la identidad del usuario a través del ingreso de su usuario clave y su clave. De acuerdo con sus funciones tendrá acceso a un perfil determinado para la realización de estas.

10.7.2.20 REQ020 – Log Out

Permite verificar la identidad del usuario a través del ingreso de su usuario clave y su clave. De acuerdo con sus funciones tendrá acceso a un perfil determinado para la realización de estas.

10.7.2.21 REQ027 – Alta Servicio

El sistema permite realizar el alta de los servicios ingresando título, descripción, precio unitario e imágenes relacionadas al mismo.

10.7.2.22 REQ028 – Modificar Servicio

El sistema permite realizar las modificaciones de los servicios ingresando título, descripción, precio unitario e imágenes relacionadas al mismo.

10.7.2.23 REQ029 – Eliminar Servicio

El sistema permite realizar la baja de servicios eliminando título, descripción, precio unitario e imágenes relacionadas al mismo.

10.7.2.24 REQ030 – Modificar Empleado

Permite al encargado de recursos humanos llevar adelante la carga de los datos relacionados a las modificaciones del empleado.

10.7.2.25 REQ031 – Eliminar Empleado

El sistema no permite eliminar a un empleado, pero si cambiarle el perfil a público y no permitirle el acceso a información sensible de la empresa.

10.7.2.26 REQ032 – Alta Dirección

El cliente puede cargar direcciones adicionales a la cargada cuando se inscribió por primera vez. Ya que un usuario puede poseer más de un domicilio donde desea contar con nuestros servicios el sistema permite realizar alta de direcciones adicionales.

10.7.2.27 REQ033 – Eliminar Dirección

El sistema permite al cliente eliminar direcciones de su lista de direcciones para servicios.

10.7.2.28 REQ034 – Alta Tarjeta

Permite al cliente llevar adelante la carga de los datos relacionados al alta de sus tarjetas ya sean de crédito o débito

10.7.2.29 REQ035 – Modificar Tarjeta

Debido a que las tarjetas ya sean de crédito o débito tienen un periodo de validez el sistema permite al cliente llevar adelante la carga de los datos relacionados a modificaciones de sus tarjetas ya sean de crédito o débito

10.7.2.30 REQ036 – Eliminar Tarjeta

Debido a que las tarjetas ya sean de crédito o débito tienen un periodo de validez el sistema permite al cliente llevar adelante la eliminación de sus tarjetas ya sean de crédito o débito

10.7.2.31 REQ037 – Recuperar Contraseña

En caso del cliente no recordar su contraseña puede realizar el proceso de recuperar contraseña donde el sistema le va a enviar por correo electrónico una nueva contraseña alfanumérica don distinción entre mayúsculas u minúsculas.

10.7.2.32 REQ043 – Genera Recibo

EL Sistema genera automáticamente cuando el usuario cliente selecciona pagar la emisión del recibo como medio de comprobante por el pago realizado.

10.7.2.33 REQ041 – Consulta Factura

El sistema permite al empleado listar las facturas y consultarlas para ver cómo están conformado su contenido.

10.7.3 Requisitos de Desempeño del Sistema

A través del sistema web se podrá acceder en todo momento a la información ya sea referente los estados y movimientos de los clientes como ordenes de trabajo, organización de la prestación de los servicios para realizar una gestión fácil de los recursos del personal para complacer las tareas pendientes de manera concreta a través de informes, sin redundancia de datos y alta velocidad.

El sistema funciona bajo las siguientes tecnologías.

Sistema Operativo Windows 10

Windows 10 es el último y vigente sistema operativo desarrollado por Microsoft como parte de la familia de sistemas operativos Windows NT.

Mac OS X

Es el sistema operativo creado por *Apple Inc*. y viene instalado en todos sus computadores. Todas las versiones recientes son conocidas como MacOS X y los nombres específicos de cada una de estas son: Mavericks, lanzada en 2013; Mountain Lion, en el 2012; Lion, en el 2011 y Snow Leopard que fue creada en el 2009.

Ubuntu

Ubuntu es un sistema operativo de software libre y código abierto. Es una distribución de Linux basada en Debian. Actualmente corre en computadores de escritorio y servidores.

Microsoft Visual Studio 2017

Es un entorno de desarrollo para sistemas operativos Windows, el cual aporta múltiples lenguajes de programación al igual que entornos de desarrollo web como ASP.NET, que es el que utilizaremos. Con este producto, los desarrolladores pueden crear aplicaciones, sitios y aplicaciones web, así como servicios web en cualquier entorno que soporte la plataforma .NET

SQL Server 2017

Para la Base de Datos de la solución utilizaremos SQL Server 2017, que es un gestor de bases de datos del tipo relacional, desarrollado por Microsoft, que nos permite crear, administrar y gestionar las tablas correspondientes al modelo de datos del negocio.

Arquitectura ASP.NET

El modelo ASP.NET es un sitio desarrollado en base a Web Forms. Web Forms son páginas que los usuarios solicitan a través de su navegador y que forman la interfaz de usuario (UI) que da su apariencia a sus aplicaciones web. Estas páginas están escritas usando una combinación de HTML, controles de servidor y código de servidor. Cuando los usuarios solicitan una página, se compila y se ejecuta en el servidor y, a continuación, genera el código HTML que el navegador puede mostrar.

Lenguaje C#

C# (pronunciado si *sharp* en inglés) es un lenguaje de programación multiparadigma de desarrollado y estandarizado por la empresa Microsoft como parte de su plataforma .NET

Navegadores

Para el funcionamiento del sistema el cliente debe contener un navegador de internet con acceso a la misma. Los más conocidos son Chrome, Opera, FireFox,

Flexible

El sistema está desarrollado de manera que permite en el futuro agregar nuevas funcionalidades y adaptarse a nuevos requisitos por parte de usuarios y tecnologías.

10.7.4 Requisitos de Entorno

Todo sistema tiene unas necesidades en la configuración de sus componentes que son indispensables para un buen funcionamiento. Dada la importancia de los datos que va a contener se precisa de rapidez y confiabilidad. Por lo tanto, para una buena performance necesitaremos Windows vista 10 o superior y vigente sistema operativo desarrollado por Microsoft como parte de la familia de sistemas operativos Windows NT.

Los usuarios podrán acceder a través de su usuario y clave a su menú correspondiente. Para tal procedimiento se debe contar con una red LAN o WAN, de acuerdo con las dimensiones de la institución, para poder conectarse con el servidor.

10.7.4.1 REQ022 – Asignar permisos a los Roles

El sistema cuenta con un alto nivel de seguridad ya que para acceder al mismo la persona poseerá usuario y contraseña brindada por personal jerárquico y solo podrán acceder a funcionalidades de acuerdo con su perfil para resguardar datos sensibles.

Por otro lado, ante las transacciones por cobranzas a través del sitio, los datos de las tarjetas de créditos deben ser encriptadas.

10.7.4.2 REQ023 – Generar Dígitos de Verificador Vertical

El objetivo del digito verificador Vertical es la suma de todos los dígitos verificadores horizontales correspondiente a una tabla determinada. Este sirve para identificar cualquier tipo de alteración, modificación no autorizada de la base de datos a través de la modificación de su contenido ya que se genera cada vez que el usuario inicia sesión.

10.7.4.3 REQ024 – BackUps

La gestión de BackUps será llevada a cabo por el Administrador se llevará a cabo manualmente de manera diaria al final del día. Los BackUps serán sobre escribidos en el momento de ejecutar el nuevo BackUps.

10.7.4.4 REQ025 – Restore

Las copias de respaldo serán guardadas en un sector seguro en el cual solo puede acceder solamente el usuario Admin ya que es información vital para el funcionamiento del sitio. El Administrador podrá ejecutar el restaurar de la base de datos, en caso de ser necesario

10.7.4.5 REQ026 – Ver Logs

El administrador podrá acceder a los registros de la bitácora filtrando por tipo de evento e indicando en qué periodo de tiempo. Para detectar altas y modificaciones en los registros por parte de los usuarios

10.7.4.6 REQ038 – Genera Digito Verificador Horizontal

El objetivo del digito verificador horizontal identificar cualquier tipo de alteración, modificación no autorizada de la base de datos. La suma de los aquellos valores indicados por el administrador correspondiente a una fila de una tabla determinada. Por ejemplo: los datos personales de un usuario. Que es almacenado en la misma fila.

10.7.4.7 REQ039 – Generar Digito Verificador Vertical

Cuando se generan modificaciones en una fila determinada, la cual genera un nuevo digito verificador horizontal, se actualiza el valor correspondiente a la tabla en la tabla de digito verificador vertical.

10.7.4.8 REQ43 – Modificar permisos a los Roles

Esta función permite al usuario Admin modificar los permisos a los roles y los permisos que tiene asignado.

10.7.4.9 REQ044 – Modificar Orden de trabajo

El sistema permite modificar la orden de trabajo ya sea cambiando su estado de activo a finalizado o los ítems relacionada a esta.

10.7.4.10 REQO45 – Consultar Roles

Esta función permite al usuario Admin y usuario Empleado ver los roles y los permisos que tiene asignado. Familia – patente.

10.10 Caso de Uso

Requisitos involucrados al sector comercial

- CU - 001 – Alta del Cliente
- CU -003 – Catalogo de Servicios
- CU - 004 – Venta mediante sitio E-Commerce
- CU - 005 – Solicitud de Presupuesto
- CU - 0006 – Cancelación de Venta de Servicio
- CU – 027 - Alta Servicio
- CU – 028 – Modificar Servicio
- CU- 029 – Eliminar Servicio
- CU – 032 – Alta Dirección
- CU – 033- Eliminar Dirección
- CU – 034 – Alta Tarjeta
- CU – 035 – Modificar Tarjeta
- CU – 036 – Eliminar Tarjeta
- CU – 037 – Recuperar Contraseña

CU - 001 - Alta del Cliente

Precondición
No existen precondiciones

Actores primarios:
Usuario cliente

Camino principal del caso de uso – Alta del Cliente

1. El usuario cliente selecciona crear una nueva cuenta.
2. El sistema solicita se ingrese:
    a. E-mail: Texto con formato ejemplo@ejemplo.com
    b. Contraseña (alfanumérica ce 8 caracteres que incluya mayúsculas y minúsculas)
    c. Reingrese contraseña.

3. El sistema solicita los demás datos básicos:
    a. Nombres o Nombre: Texto
    b. Apellidos: Texto.
    c. Sexo
    d. DNI
    e. Tipo de teléfono
    f. Número de teléfono (mínimo 8 dígitos) 1234-5678
    g. Dirección

        i. Provincia: texto
        ii. Localidad: texto
        iii. Calle: texto
        iv. Nro. de calle: Números
        v. Piso: Número.
        vi. Departamento: texto
        vii. Tipo de Domicilio

4. El usuario cliente ingresa los datos solicitados y confirma la operación crear cuenta.
5. El sistema valida los datos.
6. El sistema verifica que el usuario ya no esté registrado.
7. El sistema encripta la contraseña.
8. El sistema genera el Digito Verificador horizontal.
9. El sistema registra el nuevo usuario cliente.

10. El sistema registra en la bitácora el evento de alta de cliente.
11. El sistema actualiza el valor del Digito Verificador Vertical tabla Usuario.

Caminos alternativos

Para el paso 5: el sistema detecta que falta ingresar información o la misma no es del tipo indicado

5.1 El sistema informa que aún falta ingresar datos o los datos ingresados no son del tipo indicado.

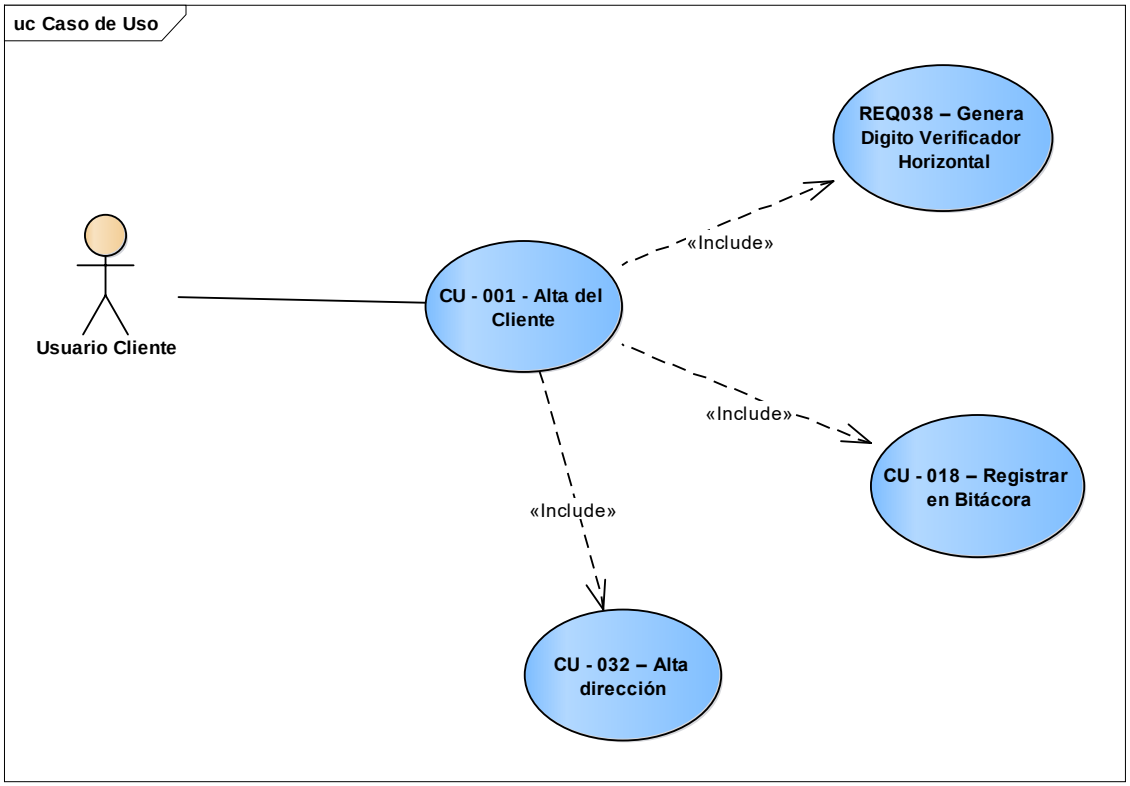
Para el paso 6: La dirección de correo electrónico ya se encuentra registrada

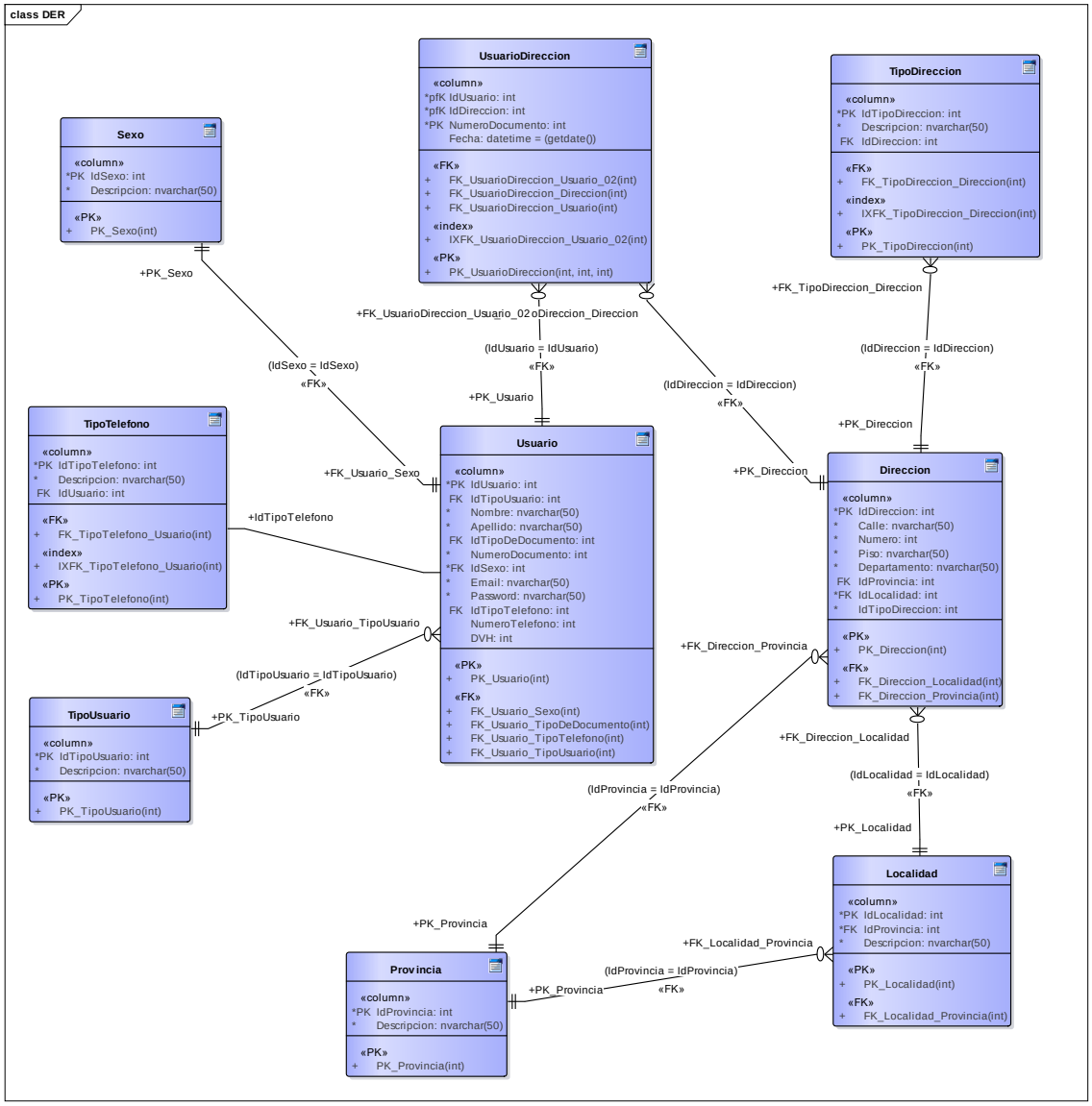
6.1 El sistema informa que la dirección de correo electrónica ya se encuentra registrada.

Postcondición
El cliente se encuentra registrado en la página.

Flujo alternativo

Falla en el momento de grabación del alta de cliente.
El sistema registra en bitácora el error producido.


uc Caso de Uso
REQ038 – Genera Digito Verificador Horizontal
«Include»
Usuario Cliente
CU - 001 - Alta del Cliente
«Include»
CU - 018 – Registrar en Bitácora
«Include»
CU - 032 – Alta dirección


class DER
Sexo
«column»
*PK IdSexo: int
* Descripcion: nvarchar(50)
«PK»
+ PK_Sexo(int)
UsuarioDireccion
«column»
*pfK IdUsuario: int
*pfK IdDireccion: int
*PK NumeroDocumento: int
Fecha: datetime = (getdate())
«FK»
+ FK_UsuarioDireccion_Usuario_02(int)
+ FK_UsuarioDireccion_Direccion(int)
+ FK_UsuarioDireccion_Usuario(int)
«index»
+ IXFK_UsuarioDireccion_Usuario_02(int)
«PK»
+ PK_UsuarioDireccion(int, int, int)
TipoDireccion
«column»
*PK IdTipoDireccion: int
* Descripcion: nvarchar(50)
FK IdDireccion: int
«FK»
+ FK_TipoDireccion_Direccion(int)
«index»
+ IXFK_TipoDireccion_Direccion(int)
«PK»
+ PK_TipoDireccion(int)
+PK_Sexo
+FK_UsuarioDireccion_Usuario_02
+FK_TipoDireccion_Direccion
(IdSexo = IdSexo)
«FK»
(IdUsuario = IdUsuario)
«FK»
(IdDireccion = IdDireccion)
«FK»
(IdDireccion = IdDireccion)
«FK»
+PK_Usuario
+PK_Direccion
TipoTelefono
«column»
*PK IdTipoTelefono: int
* Descripcion: nvarchar(50)
FK IdUsuario: int
«FK»
+ FK_TipoTelefono_Usuario(int)
«index»
+ IXFK_TipoTelefono_Usuario(int)
«PK»
+ PK_TipoTelefono(int)
+FK_Usuario_Sexo
+PK_Direccion
+IdTipoTelefono
Usuario
«column»
*PK IdUsuario: int
FK IdTipoUsuario: int
* Nombre: nvarchar(50)
* Apellido: nvarchar(50)
FK IdTipoDeDocumento: int
* NumeroDocumento: int
*FK IdSexo: int
* Email: nvarchar(50)
* Password: nvarchar(50)
FK IdTipoTelefono: int
NumeroTelefono: int
DVH: int
«PK»
+ PK_Usuario(int)
«FK»
+ FK_Usuario_Sexo(int)
+ FK_Usuario_TipoDeDocumento(int)
+ FK_Usuario_TipoTelefono(int)
+ FK_Usuario_TipoUsuario(int)
Direccion
«column»
*PK IdDireccion: int
* Calle: nvarchar(50)
* Numero: int
* Piso: nvarchar(50)
* Departamento: nvarchar(50)
FK IdProvincia: int
*FK IdLocalidad: int
* IdTipoDireccion: int
«PK»
+ PK_Direccion(int)
«FK»
+ FK_Direccion_Localidad(int)
+ FK_Direccion_Provincia(int)
+FK_Usuario_TipoUsuario
(IdTipoUsuario = IdTipoUsuario)
«FK»
+FK_Direccion_Provincia
TipoUsuario
«column»
*PK IdTipoUsuario: int
* Descripcion: nvarchar(50)
«PK»
+ PK_TipoUsuario(int)
+PK_TipoUsuario
+FK_Direccion_Localidad
(IdProvincia = IdProvincia)
«FK»
(IdLocalidad = IdLocalidad)
«FK»
+PK_Localidad
Localidad
«column»
*PK IdLocalidad: int
*FK IdProvincia: int
* Descripcion: nvarchar(50)
«PK»
+ PK_Localidad(int)
«FK»
+ FK_Localidad_Provincia(int)
+PK_Provincia
+FK_Localidad_Provincia
(IdProvincia = IdProvincia)
«FK»
+PK_Provincia
Provincia
«column»
*PK IdProvincia: int
* Descripcion: nvarchar(50)
«PK»
+ PK_Provincia(int)

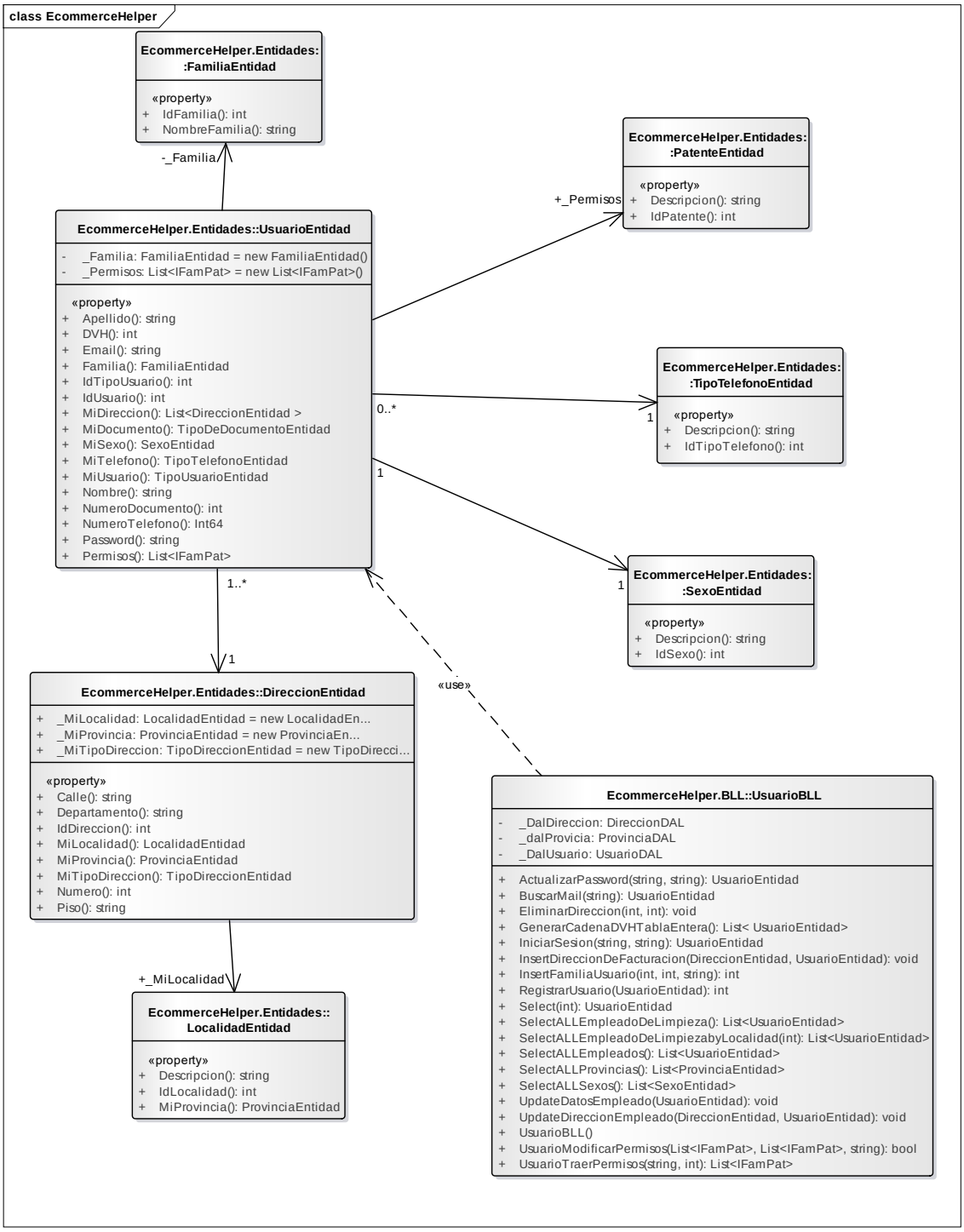
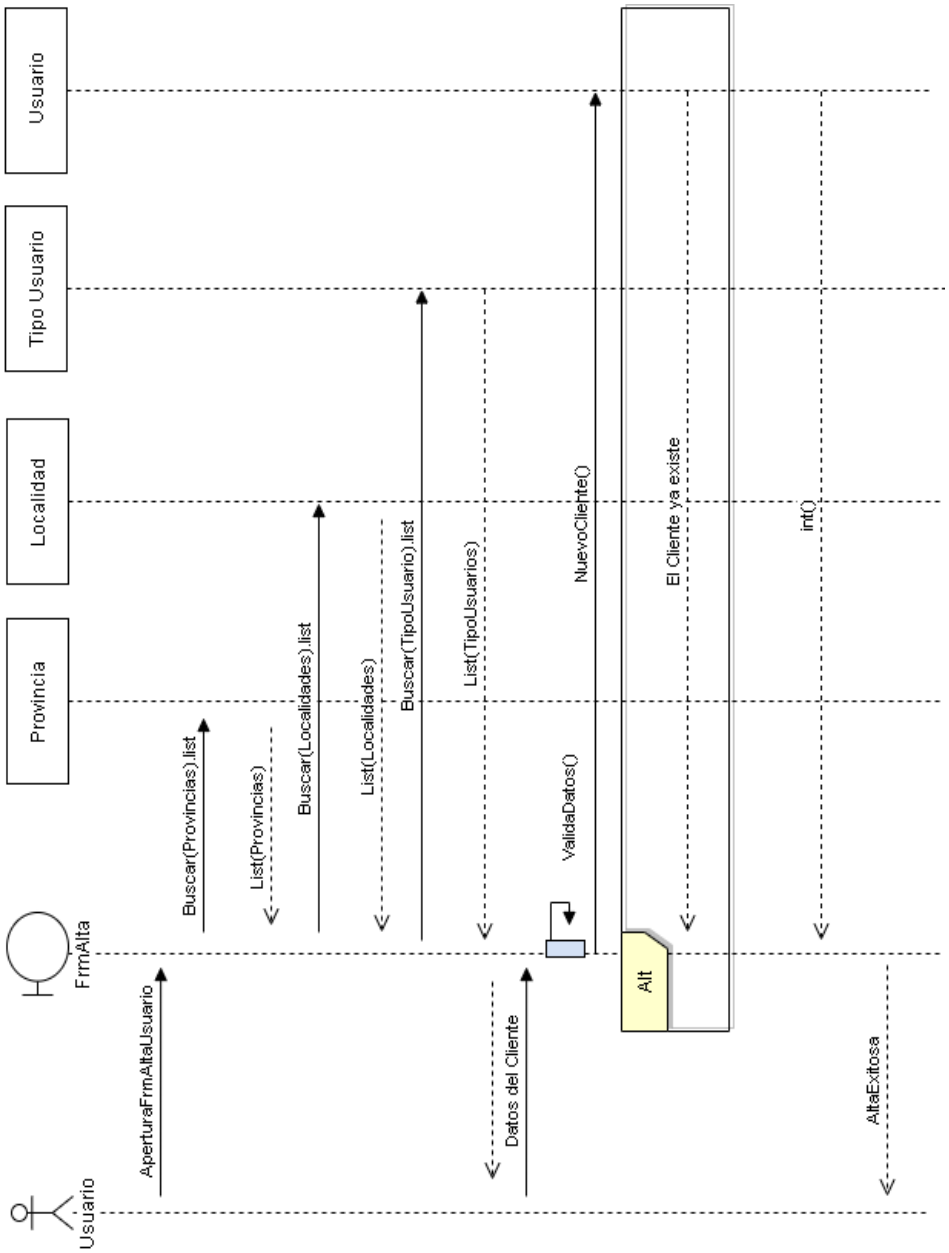
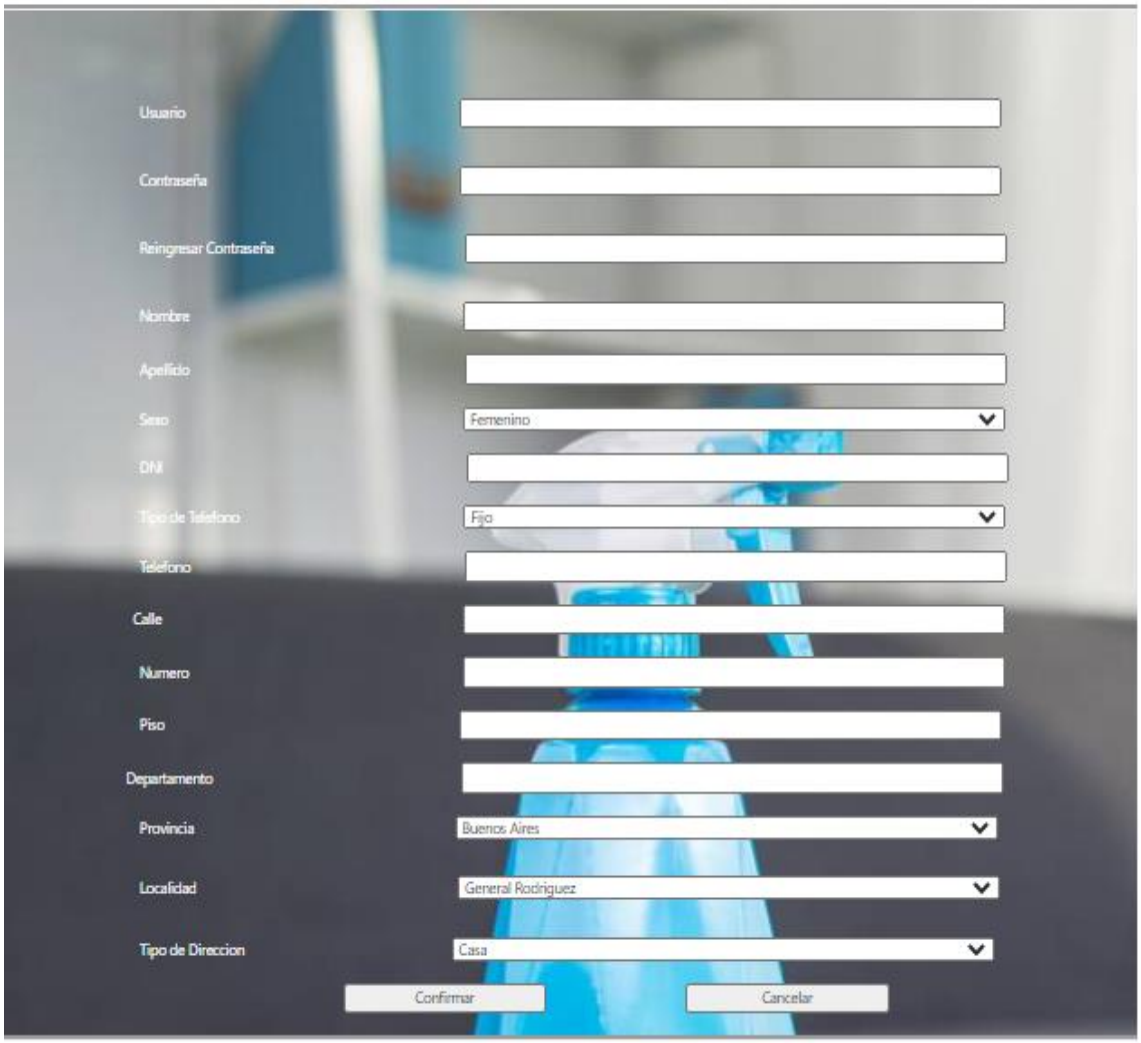

class EcommerceHelper
EcommerceHelper.Entidades::FamiliaEntidad
«property»
+ IdFamilia(): int
+ NombreFamilia(): string
-_Familia
EcommerceHelper.Entidades::PatenteEntidad
«property»
+ Descripcion(): string
+ IdPatente(): int
+_Permisos
EcommerceHelper.Entidades::UsuarioEntidad
- _Familia: FamiliaEntidad = new FamiliaEntidad()
- _Permisos: List<IFamPat> = new List<IFamPat>()
«property»
+ Apellido(): string
+ DVH(): int
+ Email(): string
+ Familia(): FamiliaEntidad
+ IdTipoUsuario(): int
+ IdUsuario(): int
+ MiDireccion(): List<DireccionEntidad >
+ MiDocumento(): TipoDeDocumentoEntidad
+ MiSexo(): SexoEntidad
+ MiTelefono(): TipoTelefonoEntidad
+ MiUsuario(): TipoUsuarioEntidad
+ Nombre(): string
+ NumeroDocumento(): int
+ NumeroTelefono(): Int64
+ Password(): string
+ Permisos(): List<IFamPat>
0..*
1
EcommerceHelper.Entidades::TipoTelefonoEntidad
«property»
+ Descripcion(): string
+ IdTipoTelefono(): int
1
EcommerceHelper.Entidades::SexoEntidad
«property»
+ Descripcion(): string
+ IdSexo(): int
1
1..*
1
EcommerceHelper.Entidades::DireccionEntidad
+ _MiLocalidad: LocalidadEntidad = new LocalidadEn...
+ _MiProvincia: ProvinciaEntidad = new ProvinciaEn...
+ _MiTipoDireccion: TipoDireccionEntidad = new TipoDirecci...
«property»
+ Calle(): string
+ Departamento(): string
+ IdDireccion(): int
+ MiLocalidad(): LocalidadEntidad
+ MiProvincia(): ProvinciaEntidad
+ MiTipoDireccion(): TipoDireccionEntidad
+ Numero(): int
+ Piso(): string
«use»
EcommerceHelper.BLL::UsuarioBLL
- _DalDireccion: DireccionDAL
- _dalProvicia: ProvinciaDAL
- _DalUsuario: UsuarioDAL
+ ActualizarPassword(string, string): UsuarioEntidad
+ BuscarMail(string): UsuarioEntidad
+ EliminarDireccion(int, int): void
+ GenerarCadenaDVHTablaEntera(): List< UsuarioEntidad>
+ IniciarSesion(string, string): UsuarioEntidad
+ InsertDireccionDeFacturacion(DireccionEntidad, UsuarioEntidad): void
+ InsertFamiliaUsuario(int, int, string): int
+ RegistrarUsuario(UsuarioEntidad): int
+ Select(int): UsuarioEntidad
+ SelectALLEmpleadoDeLimpieza(): List<UsuarioEntidad>
+ SelectALLEmpleadoDeLimpiezabyLocalidad(int): List<UsuarioEntidad>
+ SelectALLEmpleados(): List<UsuarioEntidad>
+ SelectALLProvincias(): List<ProvinciaEntidad>
+ SelectALLSexos(): List<SexoEntidad>
+ UpdateDatosEmpleado(UsuarioEntidad): void
+ UpdateDireccionEmpleado(DireccionEntidad, UsuarioEntidad): void
+ UsuarioBLL()
+ UsuarioModificarPermisos(List<IFamPat>, List<IFamPat>, string): bool
+ UsuarioTraerPermisos(string, int): List<IFamPat>
+_MiLocalidad
EcommerceHelper.Entidades::LocalidadEntidad
«property»
+ Descripcion(): string
+ IdLocalidad(): int
+ MiProvincia(): ProvinciaEntidad

Diagrama de Secuencia CU - 001 – Alta del Cliente



Pantalla

Registrarme



CU - 002 – Reporte del Cliente

Precondición
El cliente debe estar dado de alta.

Actores primarios:
Encargado - Supervisor

Camino principal del caso de uso – Reporte de Cliente

1. El usuario Encargado – Supervisor selecciona consultar Historial de consumo del Cliente.
2. El sistema solicita se ingrese:
    a. Número de DNI.
    b. Fecha de Inicio de Consulta
    c. Fecha de Finalización de Consulta.

3. El Usuario ingresa el número de DNI.
4. El sistema en base al número de DNI ingresado despliega el nombre del cliente consultado.
5. El usuario Encargado - Supervisor ingresa la fecha de inicio y la fecha de finalización del periodo y confirma la operación consultar Historial de consumos del cliente.
6. El sistema emite el reporte del cliente con los movimientos del periodo de tiempo ingresado.

Caminos Alternativos

Para el paso 2: el sistema emite un mensaje que falta el ingreso de una de las fechas solicitadas.

2.1 En caso de que el usuario Encargado – Supervisor ingrese la fechas para inicio y finalización de consulta el sistema emitirá advertencia de que falta ingreso de fecha.

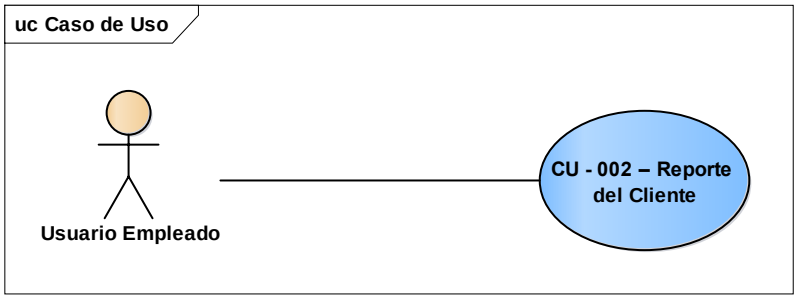
Para el paso 2: el sistema detecta que falta ingresar el número de DNI del cliente o la información ingresada es incorrecta.

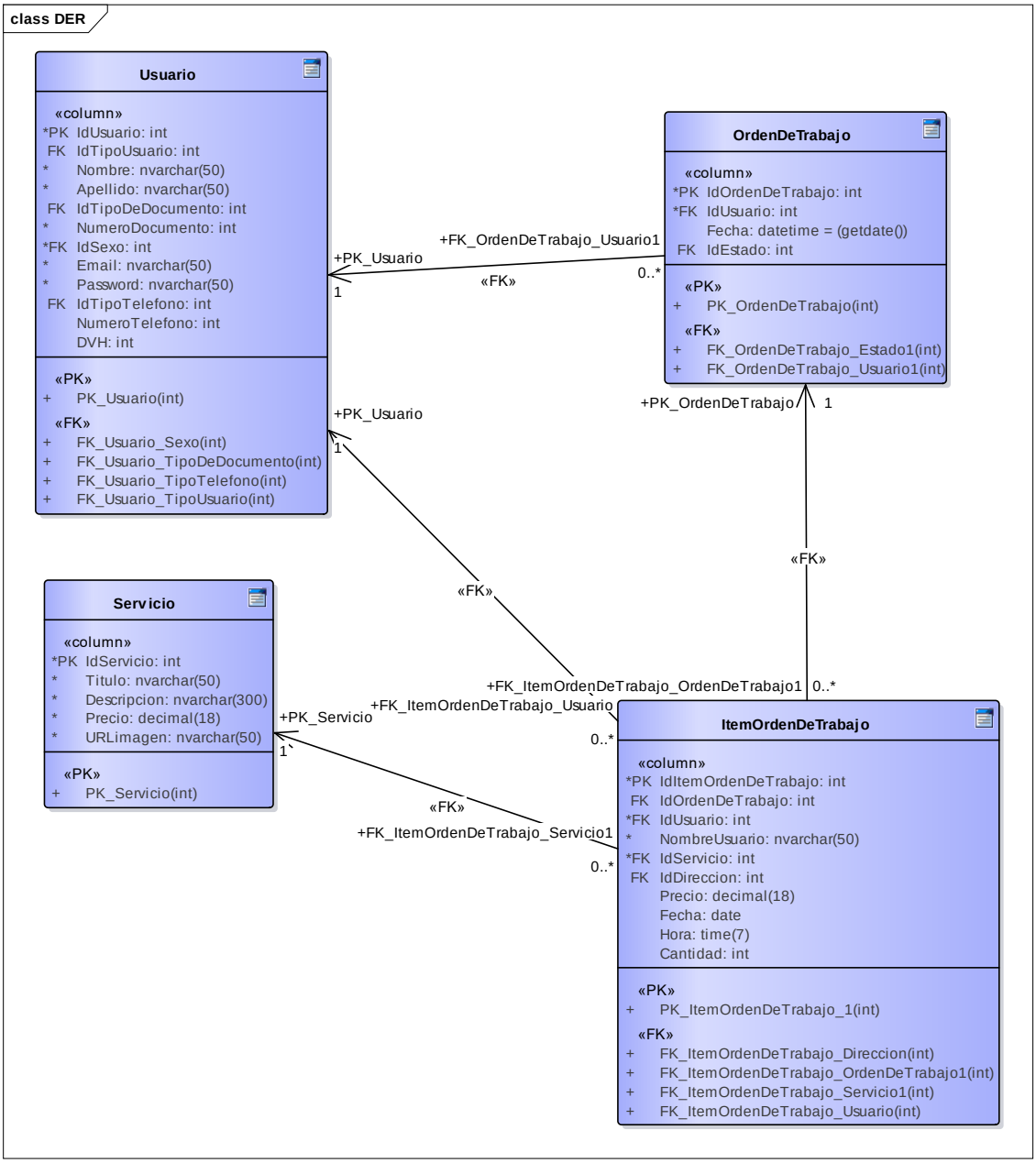
2.1 El sistema informa que aún falta ingresar datos o los datos ingresados no son del tipo indicado.

Para el paso 4: el sistema detecta que el número de DNI del cliente no está registrado

4.1 El sistema informa que el número de DNI ingresado no se encuentra registrado

Postcondición
Emisión del reporte en formato PDF.


uc Caso de Uso
Usuario Empleado
CU - 002 – Reporte del Cliente


class DER
Usuario
«column»
*PK IdUsuario: int
FK IdTipoUsuario: int
* Nombre: nvarchar(50)
* Apellido: nvarchar(50)
FK IdTipoDeDocumento: int
* NumeroDocumento: int
*FK IdSexo: int
* Email: nvarchar(50)
* Password: nvarchar(50)
FK IdTipoTelefono: int
NumeroTelefono: int
DVH: int
«PK»
+ PK_Usuario(int)
«FK»
+ FK_Usuario_Sexo(int)
+ FK_Usuario_TipoDeDocumento(int)
+ FK_Usuario_TipoTelefono(int)
+ FK_Usuario_TipoUsuario(int)
OrdenDeTrabajo
«column»
*PK IdOrdenDeTrabajo: int
*FK IdUsuario: int
Fecha: datetime = (getdate())
FK IdEstado: int
«PK»
+ PK_OrdenDeTrabajo(int)
«FK»
+ FK_OrdenDeTrabajo_Estado1(int)
+ FK_OrdenDeTrabajo_Usuario1(int)
+FK_OrdenDeTrabajo_Usuario1
+PK_Usuario
«FK»
1
0..*
+PK_Usuario
1
+PK_OrdenDeTrabajo
1
«FK»
«FK»
Servicio
«column»
*PK IdServicio: int
* Titulo: nvarchar(50)
* Descripcion: nvarchar(300)
* Precio: decimal(18)
* URLimagen: nvarchar(50)
«PK»
+ PK_Servicio(int)
+FK_ItemOrdenDeTrabajo_OrdenDeTrabajo1
0..*
+FK_ItemOrdenDeTrabajo_Usuario
+PK_Servicio
1
0..*
ItemOrdenDeTrabajo
«column»
*PK IdItemOrdenDeTrabajo: int
FK IdOrdenDeTrabajo: int
*FK IdUsuario: int
* NombreUsuario: nvarchar(50)
*FK IdServicio: int
FK IdDireccion: int
Precio: decimal(18)
Fecha: date
Hora: time(7)
Cantidad: int
«PK»
+ PK_ItemOrdenDeTrabajo_1(int)
«FK»
+ FK_ItemOrdenDeTrabajo_Direccion(int)
+ FK_ItemOrdenDeTrabajo_OrdenDeTrabajo1(int)
+ FK_ItemOrdenDeTrabajo_Servicio1(int)
+ FK_ItemOrdenDeTrabajo_Usuario(int)
«FK»
+FK_ItemOrdenDeTrabajo_Servicio1
0..*

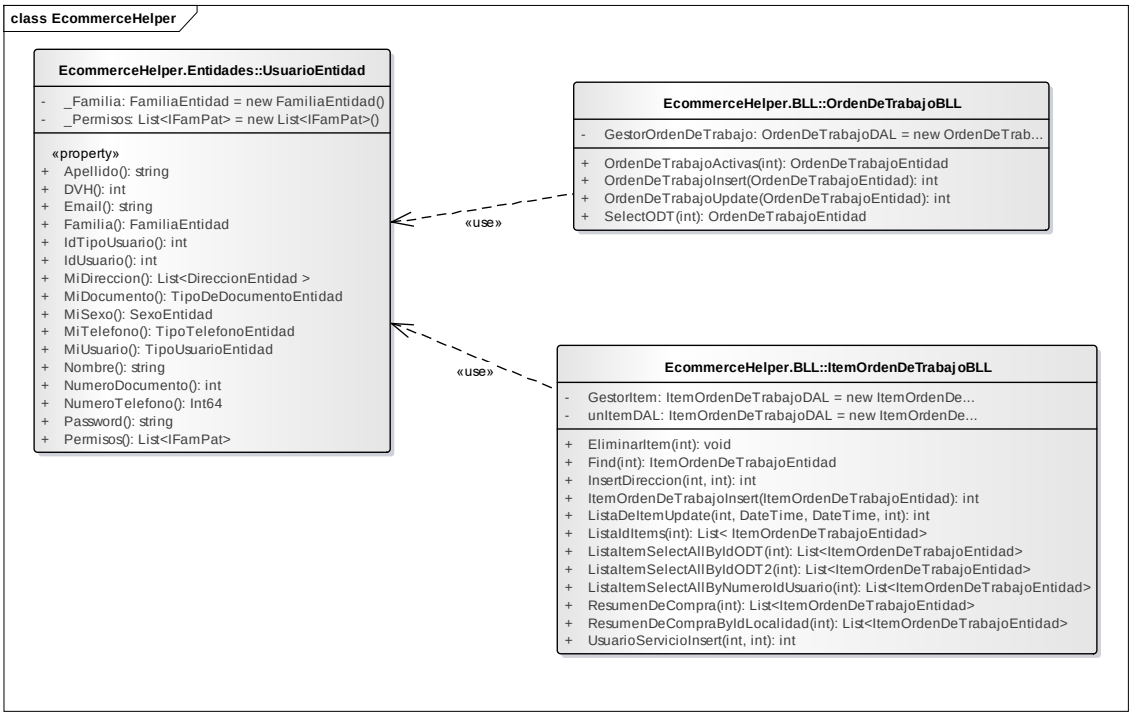
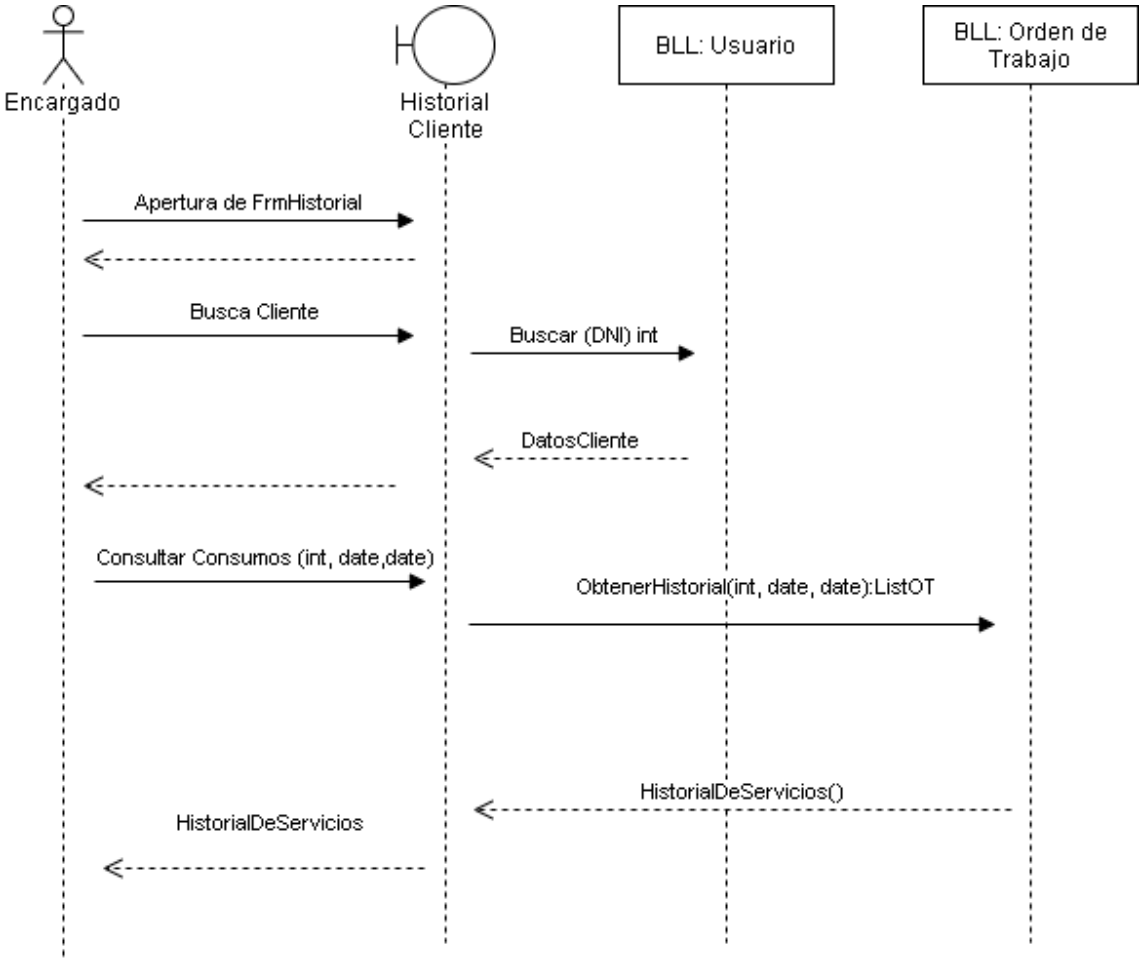

class EcommerceHelper
EcommerceHelper.Entidades::UsuarioEntidad
- _Familia: FamiliaEntidad = new FamiliaEntidad()
- _Permisos: List<IFamPat> = new List<IFamPat>()
«property»
+ Apellido(): string
+ DVH(): int
+ Email(): string
+ Familia(): FamiliaEntidad
+ IdTipoUsuario(): int
+ IdUsuario(): int
+ MiDireccion(): List<DireccionEntidad >
+ MiDocumento(): TipoDeDocumentoEntidad
+ MiSexo(): SexoEntidad
+ MiTelefono(): TipoTelefonoEntidad
+ MiUsuario(): TipoUsuarioEntidad
+ Nombre(): string
+ NumeroDocumento(): int
+ NumeroTelefono(): Int64
+ Password(): string
+ Permisos(): List<IFamPat>
«use»
EcommerceHelper.BLL::OrdenDeTrabajoBLL
- GestorOrdenDeTrabajo: OrdenDeTrabajoDAL = new OrdenDeTrab...
+ OrdenDeTrabajoActivas(int): OrdenDeTrabajoEntidad
+ OrdenDeTrabajoInsert(OrdenDeTrabajoEntidad): int
+ OrdenDeTrabajoUpdate(OrdenDeTrabajoEntidad): int
+ SelectODT(int): OrdenDeTrabajoEntidad
«use»
EcommerceHelper.BLL::ItemOrdenDeTrabajoBLL
- GestorItem: ItemOrdenDeTrabajoDAL = new ItemOrdenDe...
- unItemDAL: ItemOrdenDeTrabajoDAL = new ItemOrdenDe...
+ EliminarItem(int): void
+ Find(int): ItemOrdenDeTrabajoEntidad
+ InsertDireccion(int, int): int
+ ItemOrdenDeTrabajoInsert(ItemOrdenDeTrabajoEntidad): int
+ ListaDeItemUpdate(int, DateTime, DateTime, int): int
+ ListaIdItems(int): List< ItemOrdenDeTrabajoEntidad>
+ ListaItemSelectAllByIdODT(int): List<ItemOrdenDeTrabajoEntidad>
+ ListaItemSelectAllByIdODT2(int): List<ItemOrdenDeTrabajoEntidad>
+ ListaItemSelectAllByNumeroIdUsuario(int): List<ItemOrdenDeTrabajoEntidad>
+ ResumenDeCompra(int): List<ItemOrdenDeTrabajoEntidad>
+ ResumenDeCompraByIdLocalidad(int): List<ItemOrdenDeTrabajoEntidad>
+ UsuarioServicioInsert(int, int): int

Diagrama de Secuencia CU- 002 – Reporte del Cliente



CU - 003 – Catalogo de Servicios

Precondición:
No existe una precondición.

Actores primarios:
Cliente ò usuarios de la página en general.

Camino principal del caso de uso – Catalogo de Servicios

1. El cliente hace selecciona la opción Nuestros Servicios
2. El sistema realiza la búsqueda y despliega las opciones de promociones y paquetes.
3. El cliente selecciona el paquete de preferencia.
4. Se despliega el detalle de la opción deseada.
5. El sistema va actualizando los resultados de acuerdo con lo elegido por el cliente.

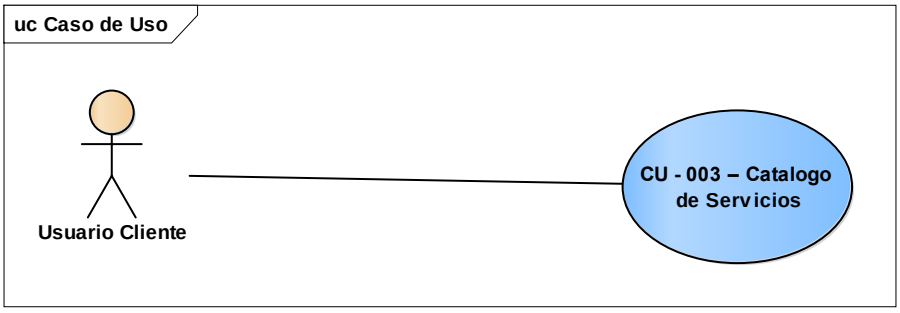
Caminos alternativos

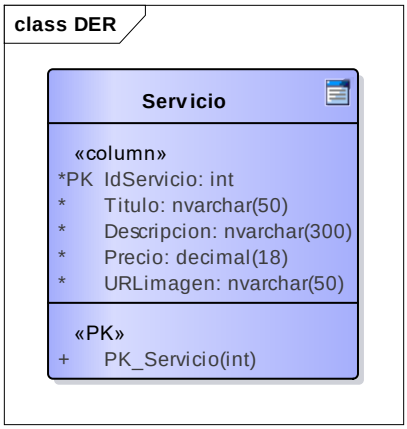
Para el paso 3:

3 a. Sin resultados para la búsqueda

3 b. El sistema informa que no se cuenta con detalle o descripción sobre el ítem solicitado.

Postcondición:
Se despliega el catálogo de servicios.


uc Caso de Uso
Usuario Cliente
CU - 003 – Catalogo de Servicios


class DER
Servicio
«column»
*PK IdServicio: int
* Titulo: nvarchar(50)
* Descripcion: nvarchar(300)
* Precio: decimal(18)
* URLimagen: nvarchar(50)
«PK»
+ PK_Servicio(int)

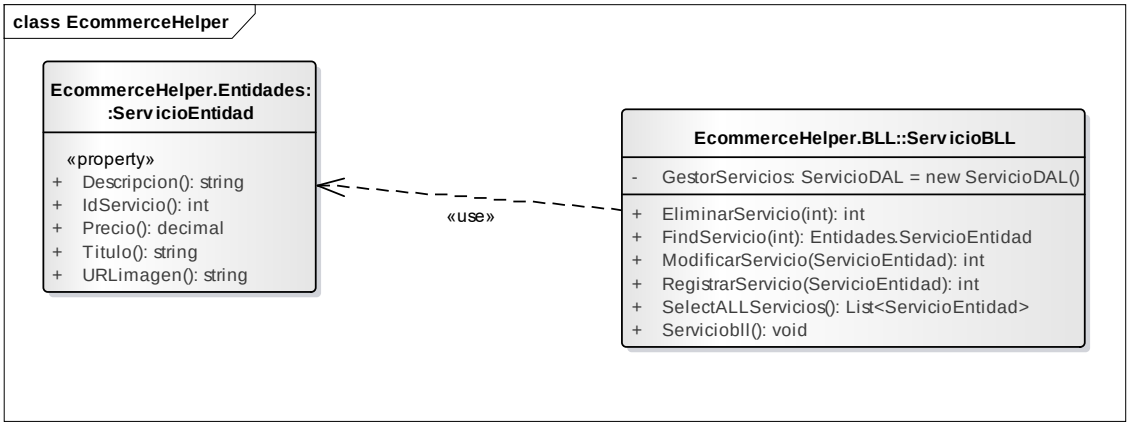
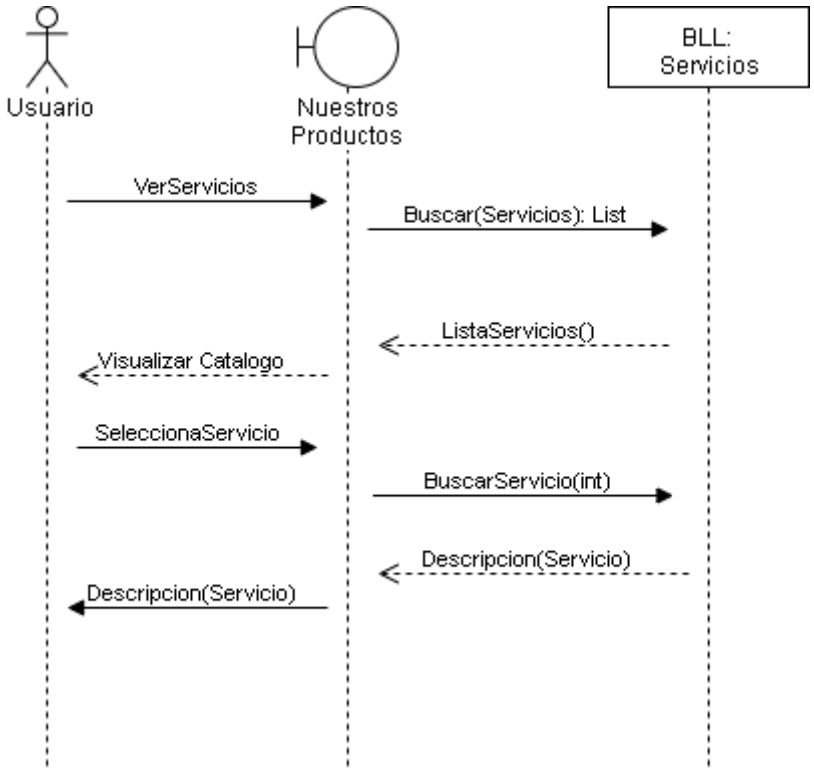
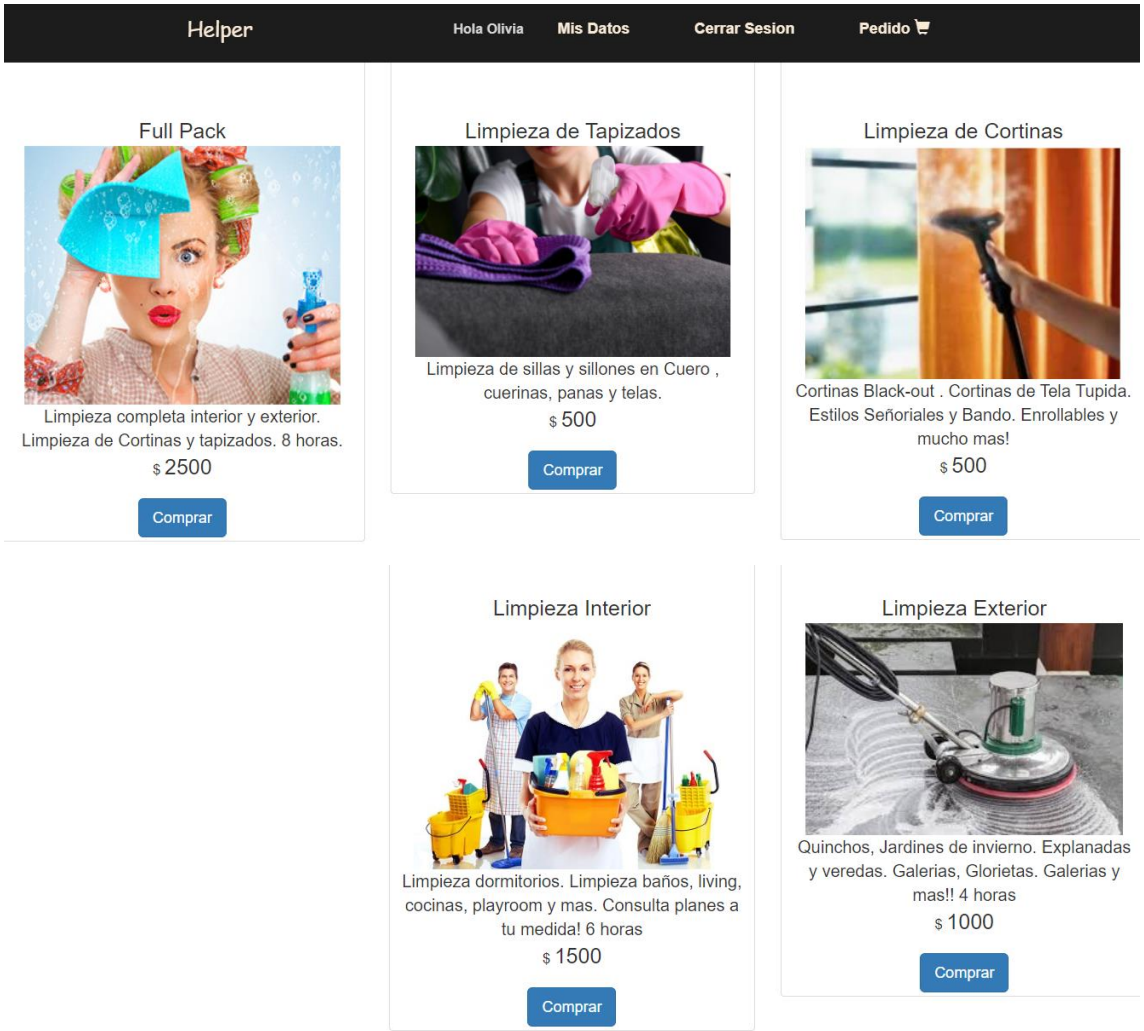

class EcommerceHelper
EcommerceHelper.Entidades::ServicioEntidad
«property»
+ Descripcion(): string
+ IdServicio(): int
+ Precio(): decimal
+ Titulo(): string
+ URLimagen(): string
«use»
EcommerceHelper.BLL::ServicioBLL
- GestorServicios: ServicioDAL = new ServicioDAL()
+ EliminarServicio(int): int
+ FindServicio(int): Entidades.ServicioEntidad
+ ModificarServicio(ServicioEntidad): int
+ RegistrarServicio(ServicioEntidad): int
+ SelectALLServicios(): List<ServicioEntidad>
+ ServiciobIl(): void

Diagrama de Secuencia CU - 003 – Catalogo de Servicios



Pantallas


Helper
Hola Olivia
Mis Datos
Cerrar Sesion
Pedido
Full Pack
Limpieza completa interior y exterior. Limpieza de Cortinas y tapizados. 8 horas.
$ 2500
Comprar
Limpieza de Tapizados
Limpieza de sillas y sillones en Cuero , cuerinas, panas y telas.
$ 500
Comprar
Limpieza de Cortinas
Cortinas Black-out . Cortinas de Tela Tupida. Estilos Señoriales y Bando. Enrollables y mucho mas!
$ 500
Comprar
Limpieza Interior
Limpieza dormitorios. Limpieza baños, living, cocinas, playroom y mas. Consulta planes a tu medida! 6 horas
$ 1500
Comprar
Limpieza Exterior
Quinchos, Jardines de invierno. Explanadas y veredas. Galerias, Glorietas. Galerias y mas!! 4 horas
$ 1000
Comprar

CU - 004 – Venta mediante sitio E-Commerce

Precondición:
El usuario cliente debe estar logueado.

Actores primarios:
Cliente

Camino principal del caso de uso - Realizar una venta on-line

1. El cliente revisa el catálogo de venta de los servicios CU-003 – Catalogo de servicios.
2. El cliente selecciona del catálogo un Servicios para ver sus detalles.
3. El cliente selecciona agregar el servicio a pedido.
4. El cliente debe ingresar para los servicios solicitados fecha, hora y cantidad.
5. El sistema valida que los datos de fecha, hora y cantidad hayan sido ingresados.
6. El cliente selecciona continuar a direcciones.
7. El sistema despliega las direcciones cargadas en la base de datos del cliente.
8. El cliente selecciona una dirección.
9. El cliente se dirige a resumen de compra.
10. El sistema despliega los servicios contratados, fecha, hora, cantidad, precio unitario, precio total y dirección en la que se va a realizar el servicio.
11. El cliente selecciona pagar.
12. El sistema solicita se seleccione la forma de pago CU -016- Cobranza.
13. El sistema genera la factura CU – 013 - Facturación.
14. El sistema genera el recibo, comprobante de pago CU – 044 – Genera Recibo.
15. El sistema actualiza el estado de la orden de trabajo ha finalizado.
16. El sistema muestra cartel de pago exitoso.

Caminos Alternativos

Para el paso 4: El cliente selecciona otro servicio para agregarlo a la orden de pedido.

4.1 El sistema agrega dicho servicio a la Orden de Pedido y muestra en una grilla el servicio fue agregado.

Para el paso 9: El cliente desea ingresar una nueva dirección.

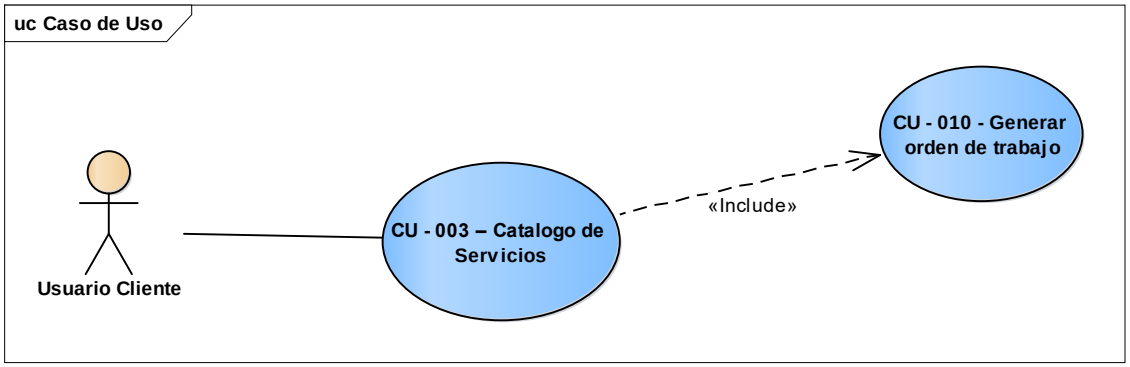
9. 1 selecciona CU- 032- Alta dirección.

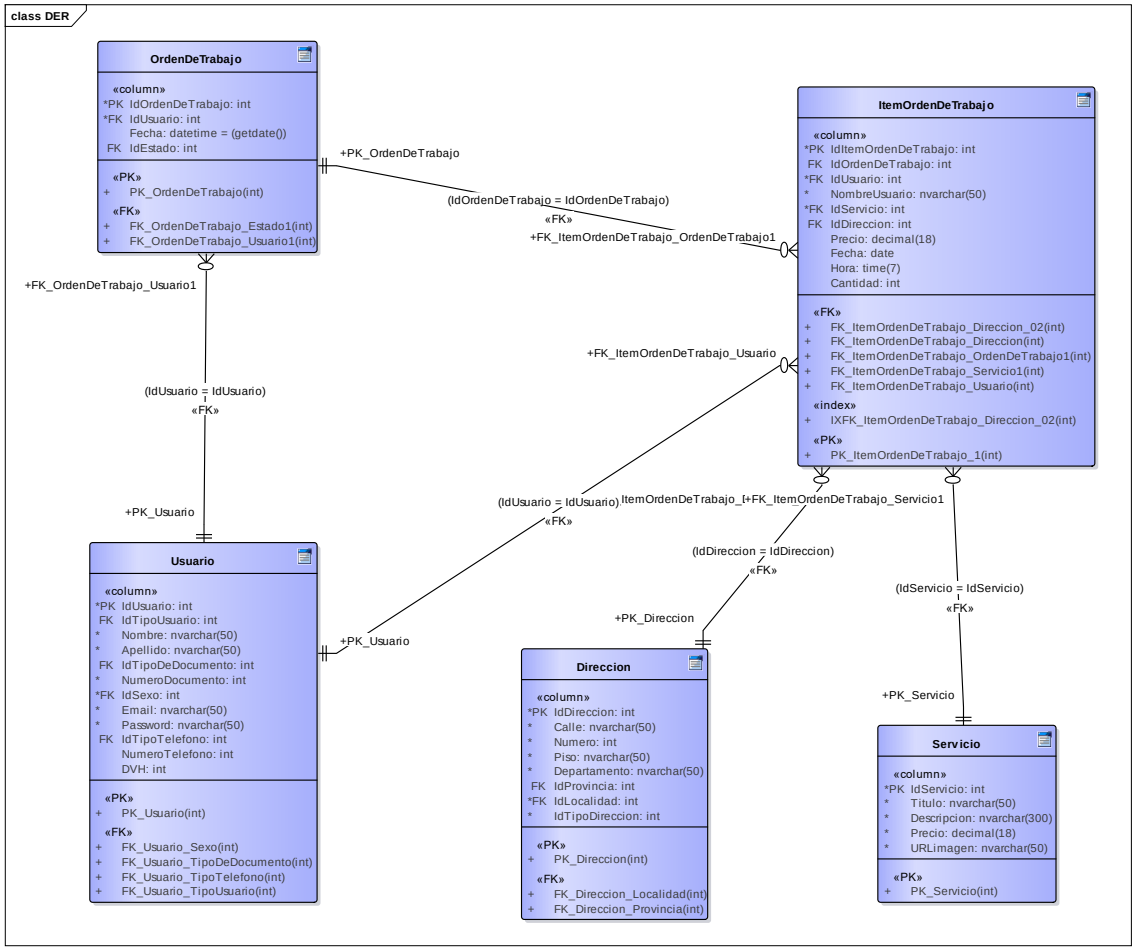
Para el paso 17: Proceso de pago incorrecto.

17.1 el proceso de pago fue incorrecto o erróneo.

Postcondición:

Venta on-line efectuada, Orden de Pedido on-line generada, confección de factura y registro de pago en recibo.


uc Caso de Uso
Usuario Cliente
CU - 003 – Catalogo de Servicios
«Include»
CU - 010 - Generar orden de trabajo


class DER
OrdenDeTrabajo
«column»
*PK IdOrdenDeTrabajo: int
*FK IdUsuario: int
Fecha: datetime = (getdate())
FK IdEstado: int
«PK»
+ PK_OrdenDeTrabajo(int)
«FK»
+ FK_OrdenDeTrabajo_Estado1(int)
+ FK_OrdenDeTrabajo_Usuario1(int)
+PK_OrdenDeTrabajo
(IdOrdenDeTrabajo = IdOrdenDeTrabajo)
«FK»
+FK_ItemOrdenDeTrabajo_OrdenDeTrabajo1
ItemOrdenDeTrabajo
«column»
*PK IdItemOrdenDeTrabajo: int
FK IdOrdenDeTrabajo: int
*FK IdUsuario: int
* NombreUsuario: nvarchar(50)
*FK IdServicio: int
FK IdDireccion: int
Precio: decimal(18)
Fecha: date
Hora: time(7)
Cantidad: int
«FK»
+ FK_ItemOrdenDeTrabajo_Direccion_02(int)
+ FK_ItemOrdenDeTrabajo_Direccion(int)
+ FK_ItemOrdenDeTrabajo_OrdenDeTrabajo1(int)
+ FK_ItemOrdenDeTrabajo_Servicio1(int)
+ FK_ItemOrdenDeTrabajo_Usuario(int)
«index»
+ IXFK_ItemOrdenDeTrabajo_Direccion_02(int)
«PK»
+ PK_ItemOrdenDeTrabajo_1(int)
+FK_OrdenDeTrabajo_Usuario1
(IdUsuario = IdUsuario)
«FK»
+FK_ItemOrdenDeTrabajo_Usuario
+PK_Usuario
(IdUsuario = IdUsuario)
«FK»
ItemOrdenDeTrabajo_[+FK_ItemOrdenDeTrabajo_Servicio1
(IdDireccion = IdDireccion)
«FK»
(IdServicio = IdServicio)
«FK»
+PK_Direccion
+PK_Usuario
+PK_Servicio
Usuario
«column»
*PK IdUsuario: int
FK IdTipoUsuario: int
* Nombre: nvarchar(50)
* Apellido: nvarchar(50)
FK IdTipoDeDocumento: int
* NumeroDocumento: int
*FK IdSexo: int
* Email: nvarchar(50)
* Password: nvarchar(50)
FK IdTipoTelefono: int
NumeroTelefono: int
DVH: int
«PK»
+ PK_Usuario(int)
«FK»
+ FK_Usuario_Sexo(int)
+ FK_Usuario_TipoDeDocumento(int)
+ FK_Usuario_TipoTelefono(int)
+ FK_Usuario_TipoUsuario(int)
Direccion
«column»
*PK IdDireccion: int
* Calle: nvarchar(50)
* Numero: int
* Piso: nvarchar(50)
* Departamento: nvarchar(50)
FK IdProvincia: int
*FK IdLocalidad: int
* IdTipoDireccion: int
«PK»
+ PK_Direccion(int)
«FK»
+ FK_Direccion_Localidad(int)
+ FK_Direccion_Provincia(int)
Servicio
«column»
*PK IdServicio: int
* Titulo: nvarchar(50)
* Descripcion: nvarchar(300)
* Precio: decimal(18)
* URLimagen: nvarchar(50)
«PK»
+ PK_Servicio(int)

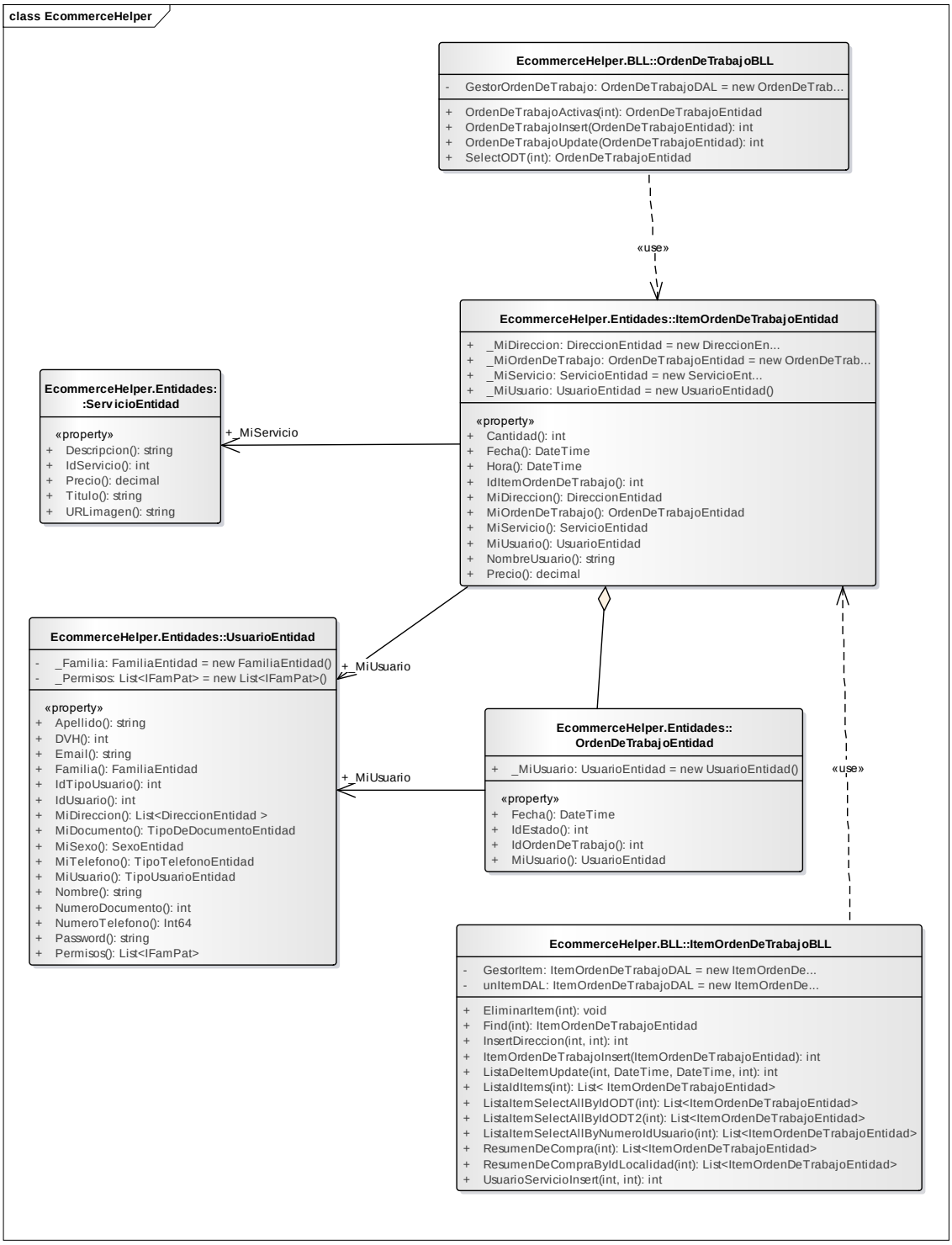
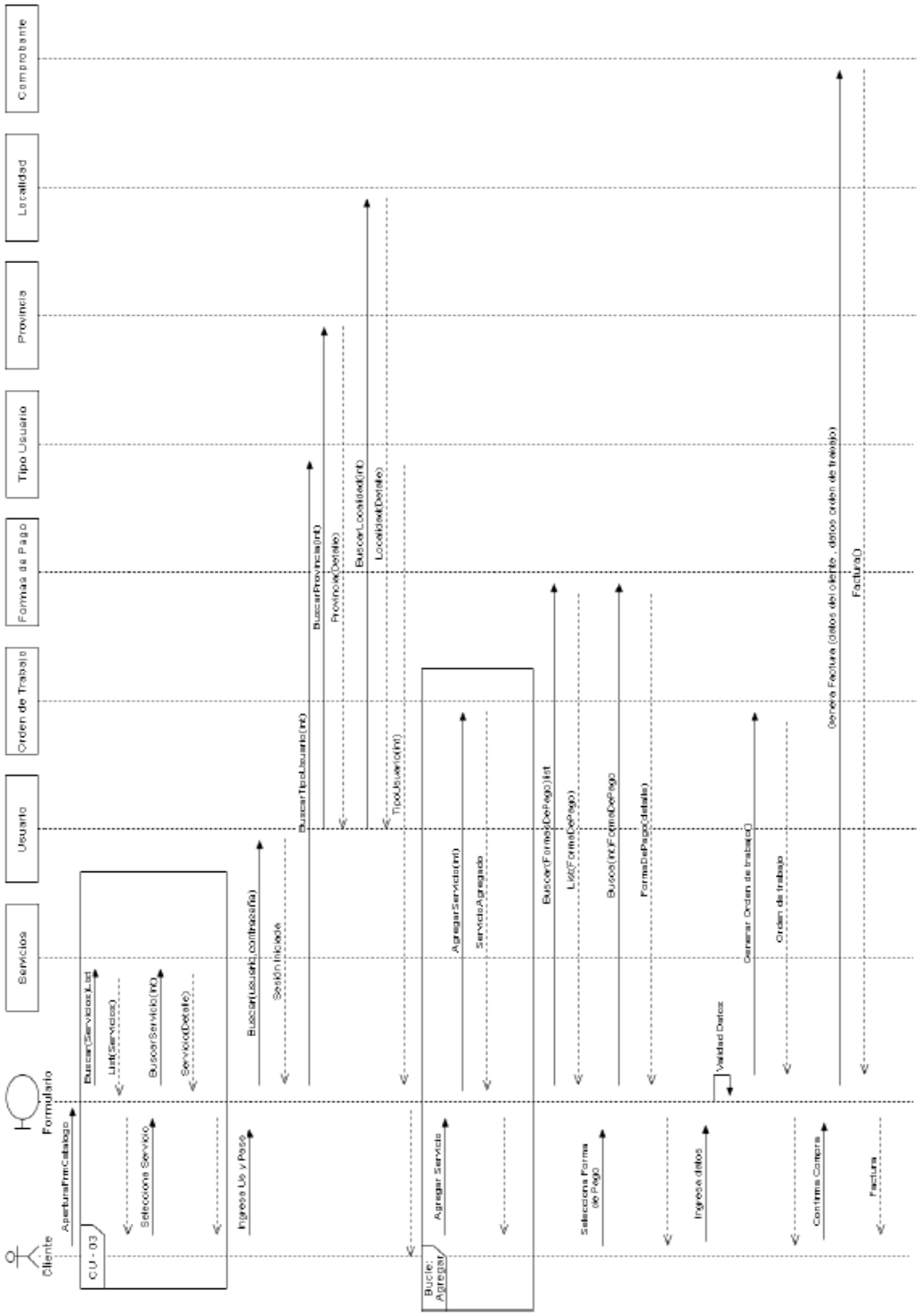
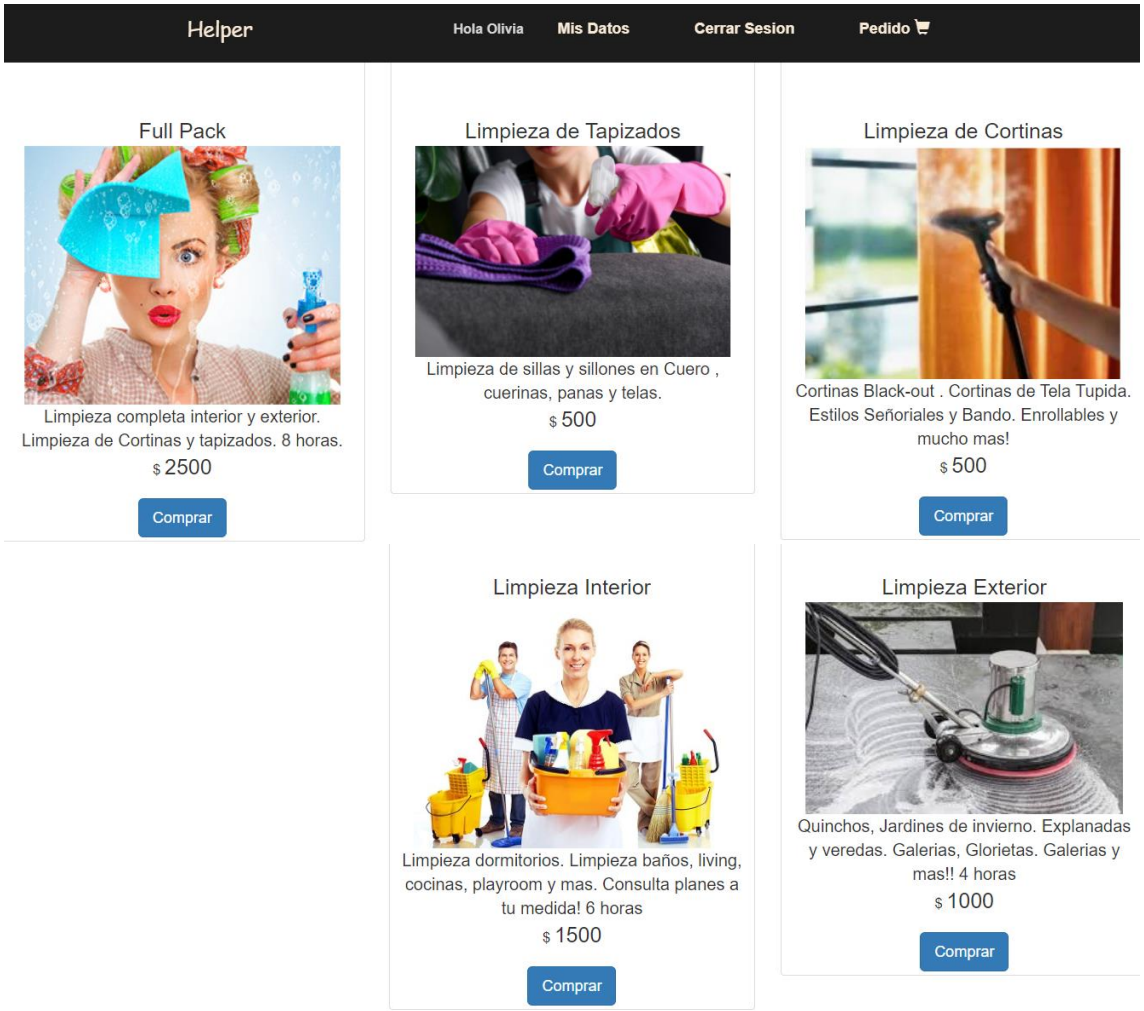

class EcommerceHelper
EcommerceHelper.BLL::OrdenDeTrabajoBLL
- GestorOrdenDeTrabajo: OrdenDeTrabajoDAL = new OrdenDeTrab...
+ OrdenDeTrabajoActivas(int): OrdenDeTrabajoEntidad
+ OrdenDeTrabajoInsert(OrdenDeTrabajoEntidad): int
+ OrdenDeTrabajoUpdate(OrdenDeTrabajoEntidad): int
+ SelectODT(int): OrdenDeTrabajoEntidad
«use»
EcommerceHelper.Entidades::ItemOrdenDeTrabajoEntidad
+ _MiDireccion: DireccionEntidad = new DireccionEn...
+ _MiOrdenDeTrabajo: OrdenDeTrabajoEntidad = new OrdenDeTrab...
+ _MiServicio: ServicioEntidad = new ServicioEnt...
+ _MiUsuario: UsuarioEntidad = new UsuarioEntidad()
«property»
+ Cantidad(): int
+ Fecha(): DateTime
+ Hora(): DateTime
+ IdItemOrdenDeTrabajo(): int
+ MiDireccion(): DireccionEntidad
+ MiOrdenDeTrabajo(): OrdenDeTrabajoEntidad
+ MiServicio(): ServicioEntidad
+ MiUsuario(): UsuarioEntidad
+ NombreUsuario(): string
+ Precio(): decimal
EcommerceHelper.Entidades::ServicioEntidad
«property»
+ Descripcion(): string
+ IdServicio(): int
+ Precio(): decimal
+ Titulo(): string
+ URLimagen(): string
+ _MiServicio
EcommerceHelper.Entidades::UsuarioEntidad
- _Familia: FamiliaEntidad = new FamiliaEntidad()
- _Permisos: List<IFamPat> = new List<IFamPat>()
«property»
+ Apellido(): string
+ DVH(): int
+ Email(): string
+ Familia(): FamiliaEntidad
+ IdTipoUsuario(): int
+ IdUsuario(): int
+ MiDireccion(): List<DireccionEntidad >
+ MiDocumento(): TipoDeDocumentoEntidad
+ MiSexo(): SexoEntidad
+ MiTelefono(): TipoTelefonoEntidad
+ MiUsuario(): TipoUsuarioEntidad
+ Nombre(): string
+ NumeroDocumento(): int
+ NumeroTelefono(): Int64
+ Password(): string
+ Permisos(): List<IFamPat>
+ _MiUsuario
+ _MiUsuario
EcommerceHelper.Entidades::OrdenDeTrabajoEntidad
+ _MiUsuario: UsuarioEntidad = new UsuarioEntidad()
«property»
+ Fecha(): DateTime
+ IdEstado(): int
+ IdOrdenDeTrabajo(): int
+ MiUsuario(): UsuarioEntidad
«use»
EcommerceHelper.BLL::ItemOrdenDeTrabajoBLL
- GestorItem: ItemOrdenDeTrabajoDAL = new ItemOrdenDe...
- unItemDAL: ItemOrdenDeTrabajoDAL = new ItemOrdenDe...
+ EliminarItem(int): void
+ Find(int): ItemOrdenDeTrabajoEntidad
+ InsertDireccion(int, int): int
+ ItemOrdenDeTrabajoInsert(ItemOrdenDeTrabajoEntidad): int
+ ListaDeItemUpdate(int, DateTime, DateTime, int): int
+ ListaIdItems(int): List< ItemOrdenDeTrabajoEntidad>
+ ListaItemSelectAllByIdODT(int): List<ItemOrdenDeTrabajoEntidad>
+ ListaItemSelectAllByIdODT2(int): List<ItemOrdenDeTrabajoEntidad>
+ ListaItemSelectAllByNumeroIdUsuario(int): List<ItemOrdenDeTrabajoEntidad>
+ ResumenDeCompra(int): List<ItemOrdenDeTrabajoEntidad>
+ ResumenDeCompraByIdLocalidad(int): List<ItemOrdenDeTrabajoEntidad>
+ UsuarioServicioInsert(int, int): int

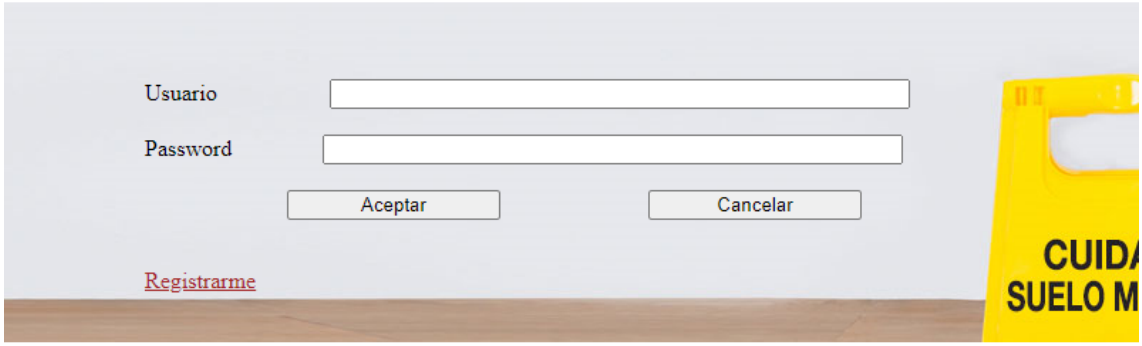
Diagrama de Secuencia CU - 004 – Venta mediante sitio E-Commerce



Pantallas


Helper
Hola Olivia
Mis Datos
Cerrar Sesion
Pedido
Full Pack
Limpieza completa interior y exterior. Limpieza de Cortinas y tapizados. 8 horas.
$ 2500
Comprar
Limpieza de Tapizados
Limpieza de sillas y sillones en Cuero , cuerinas, panas y telas.
$ 500
Comprar
Limpieza de Cortinas
Cortinas Black-out . Cortinas de Tela Tupida. Estilos Señoriales y Bando. Enrollables y mucho mas!
$ 500
Comprar
Limpieza Interior
Limpieza dormitorios. Limpieza baños, living, cocinas, playroom y mas. Consulta planes a tu medida! 6 horas
$ 1500
Comprar
Limpieza Exterior
Quinchos, Jardines de invierno. Explanadas y veredas. Galerias, Glorietas. Galerias y mas!! 4 horas
$ 1000
Comprar

Iniciar Sesion



CopyRight Argentina 2020 - Helper

Titulo	Descripcion	Precio	Imagen	Fecha	Hora	Cantidad	
Limpieza Exterior	Quinchos, Jardines de invierno. Explanadas y veredas. Galerias, Glorietas. Galerias y mas!! 4 horas	1000		30/11/2020	08:00	1	Eliminar

Calle	Numero	Piso	Departamento	Localidad	Provincia	Tipo Domicilio		
avellaneda	1078	0	0	La Reja	Buenos Aires	Quinta	Seleccionar	Eliminar
Int. Luis Tulissi	2348	1	a	Moreno	Buenos Aires	Departamento	Seleccionar	Eliminar

Agregar Direccion Ver Resumen de Compra

Titulo	Descripcion		Fecha	Hora	Cantidad	Precio
Limpieza Exterior	Quinchos, Jardines de invierno. Explanadas y veredas. Galerias, Glorietas. Galerias y mas!! 4 horas		30/11/2020	10:00	1	1000

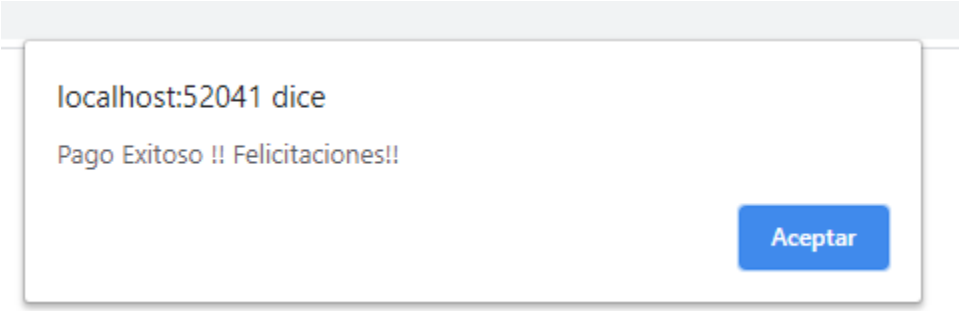
Total $ 1000

Calle	Numero	Piso	Departamento	Localidad	Provincia	Tipo Domicilio
avellaneda	1078	0	0	La Reja	Buenos Aires	Quinta
Int. Luis Tulissi	2348	1	a	Moreno	Buenos Aires	Departamento

Pagar Cancelar

Titular	Numero	Tipo Tarjeta	Fecha de Vencimiento	Codigo de Seguridad	
Esteban Ensueño Pesadilla	4398189693448976	Credito	30/11/2020	111	Seleccionar

Registrar / Actualizar Tarjeta



CU - 005 – Solicitud de Presupuesto

Precondición:
No existen precondiciones

Actores primarios:
Cliente

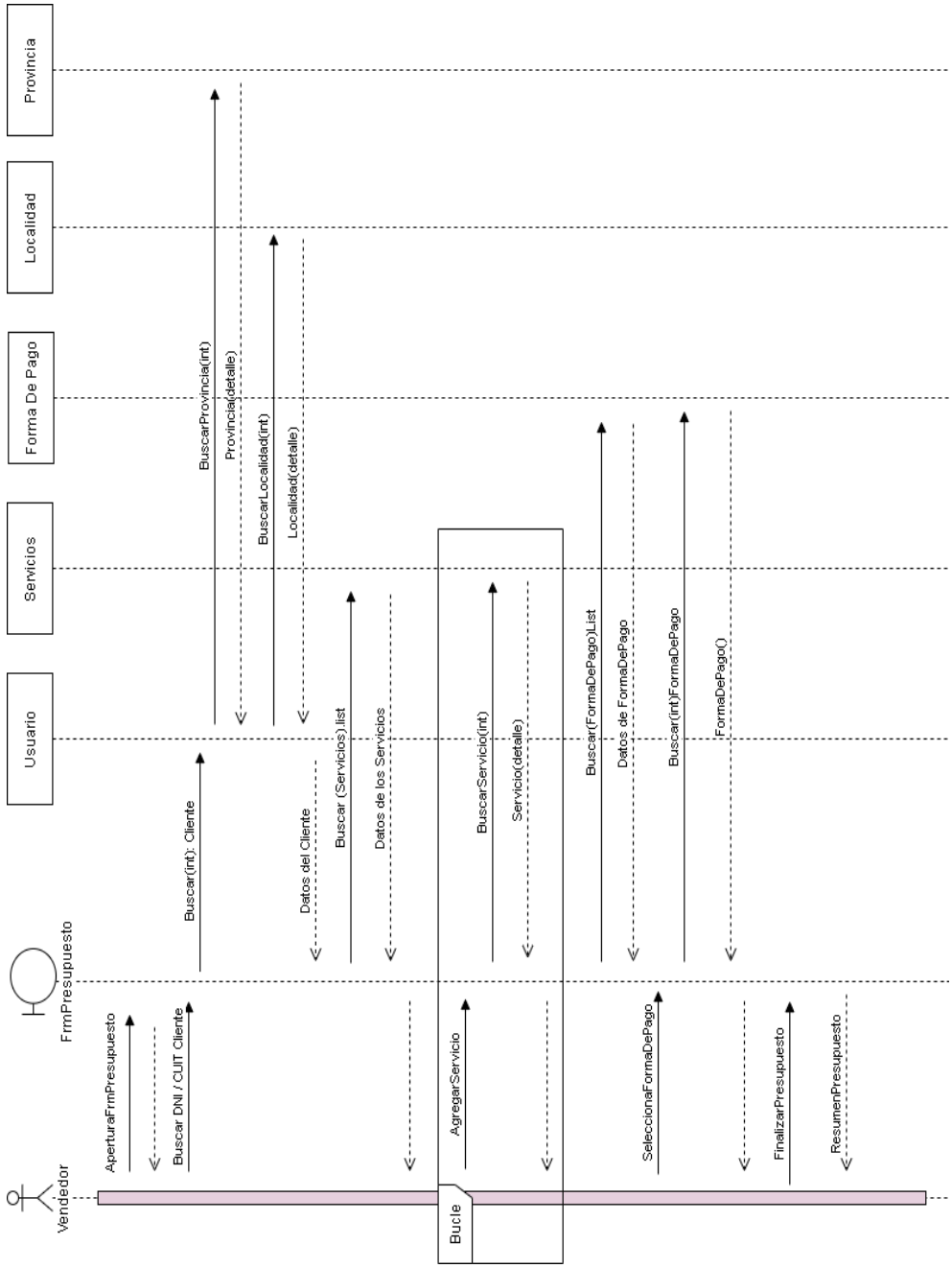
Camino principal del caso de uso - Confeccionar un presupuesto

1. El Cliente selecciona la opción para solicitar un Presupuesto
2. El sistema le solicita al cliente:
    a. Nombre y Apellido: Texto
    b. Domicilio:
        i. Calle: texto
        ii. Nro. de calle: Números
        iii. Piso y Dpto.: Número y texto
        iv. Entrecalles: texto
        v. Localidad
        vi. Provincia
    c. E-mail
3. El cliente agrega el servicio o los servicios que desea solicitar.
4. El sistema valida los datos.
5. El sistema envía los datos cargados en modo correo electrónico a la casilla de mail de Helper.

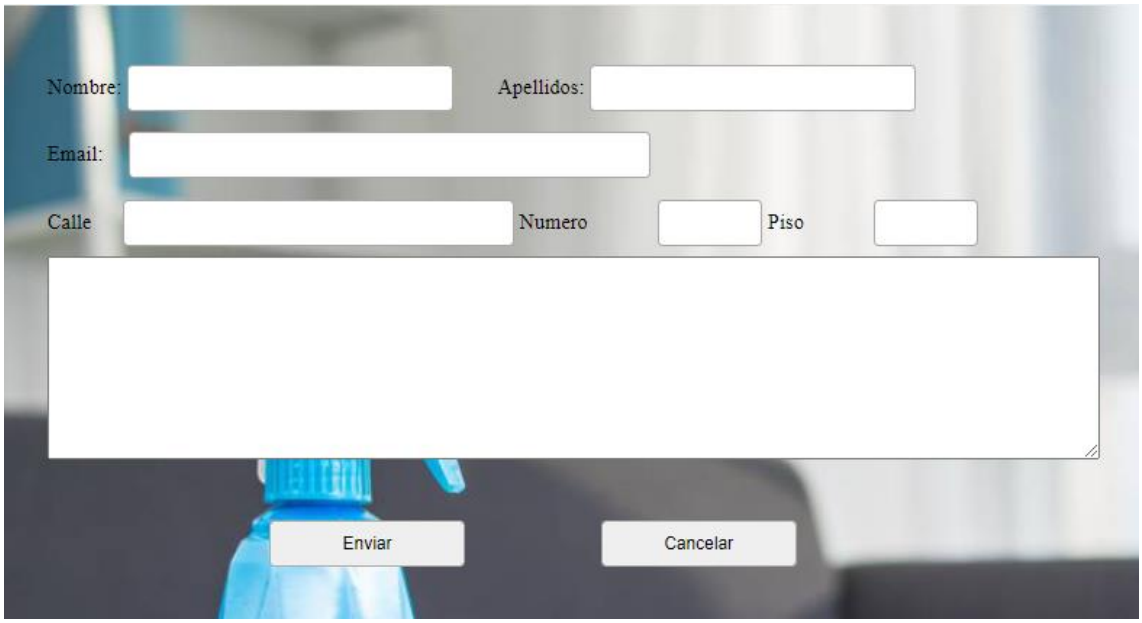
Postcondición:
La solicitud de presupuesto solicitado fue enviado a la casilla de correo.

uc Caso de Uso
Usuario Cliente
CU - 005 – Solicitud de Presupuesto

Diagrama de Secuencia CU - 005 – Solicitud de Presupuesto



Pantalla



CU - 006 – Cancelación de Venta

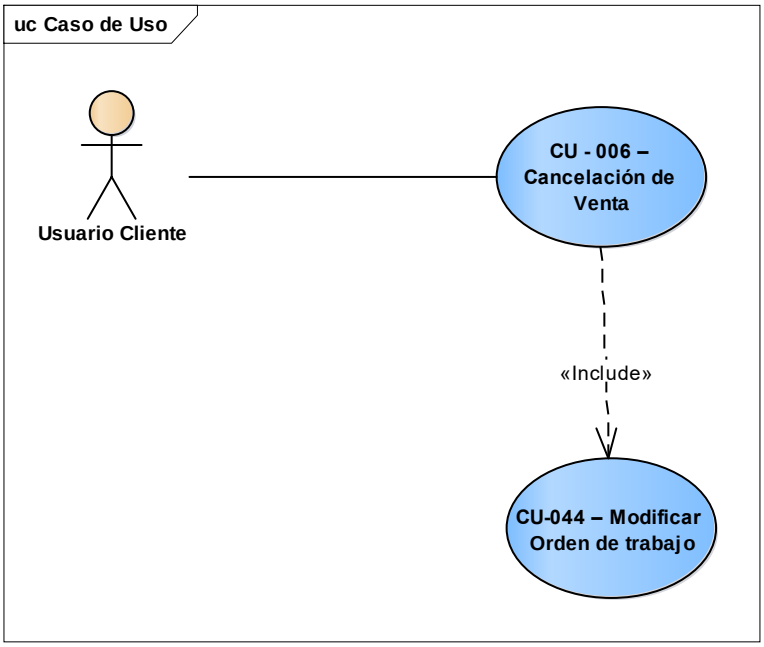
Precondición:
Sesión de usuario iniciada y debe existir al menos un servicio contratado pendiente.

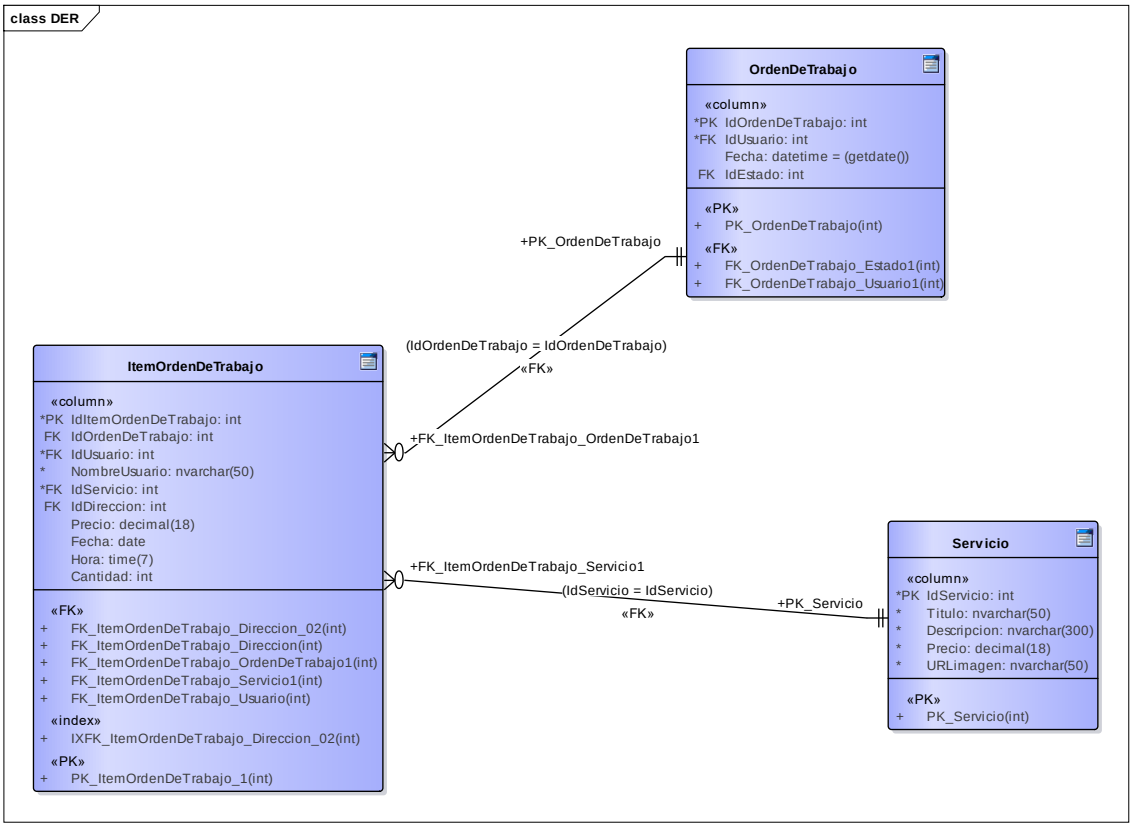
Actores primarios:
Cliente

Camino principal del caso de uso – Cancelación de Venta

1. El cliente selecciona “pedido”
2. El sistema resuelve la consulta y carga la grilla.
3. El sistema carga las compras de servicios realizadas que aún no han sido realizadas. Muestra fecha de servicio, cantidad, descripción del servicio e importe.
4. El cliente selecciona el ítem a solicitar la cancelación.
5. El sistema elimina el ítem seleccionado.
6. El sistema resuelve la modificación y refresca la lista de compras.

Postcondición:
Eliminación del servicio.


uc Caso de Uso
Usuario Cliente
CU - 006 – Cancelación de Venta
«Include»
CU-044 – Modificar Orden de trabajo


class DER
OrdenDeTrabajo
«column»
*PK IdOrdenDeTrabajo: int
*FK IdUsuario: int
Fecha: datetime = (getdate())
FK IdEstado: int
«PK»
+ PK_OrdenDeTrabajo(int)
«FK»
+ FK_OrdenDeTrabajo_Estado1(int)
+ FK_OrdenDeTrabajo_Usuario1(int)
+PK_OrdenDeTrabajo
(IdOrdenDeTrabajo = IdOrdenDeTrabajo)
«FK»
ItemOrdenDeTrabajo
«column»
*PK IdItemOrdenDeTrabajo: int
FK IdOrdenDeTrabajo: int
*FK IdUsuario: int
* NombreUsuario: nvarchar(50)
*FK IdServicio: int
FK IdDireccion: int
Precio: decimal(18)
Fecha: date
Hora: time(7)
Cantidad: int
«FK»
+ FK_ItemOrdenDeTrabajo_Direccion_02(int)
+ FK_ItemOrdenDeTrabajo_Direccion(int)
+ FK_ItemOrdenDeTrabajo_OrdenDeTrabajo1(int)
+ FK_ItemOrdenDeTrabajo_Servicio1(int)
+ FK_ItemOrdenDeTrabajo_Usuario(int)
«index»
+ IXFK_ItemOrdenDeTrabajo_Direccion_02(int)
«PK»
+ PK_ItemOrdenDeTrabajo_1(int)
+FK_ItemOrdenDeTrabajo_OrdenDeTrabajo1
+FK_ItemOrdenDeTrabajo_Servicio1
(IdServicio = IdServicio)
«FK»
+PK_Servicio
Servicio
«column»
*PK IdServicio: int
* Titulo: nvarchar(50)
* Descripcion: nvarchar(300)
* Precio: decimal(18)
* URLimagen: nvarchar(50)
«PK»
+ PK_Servicio(int)

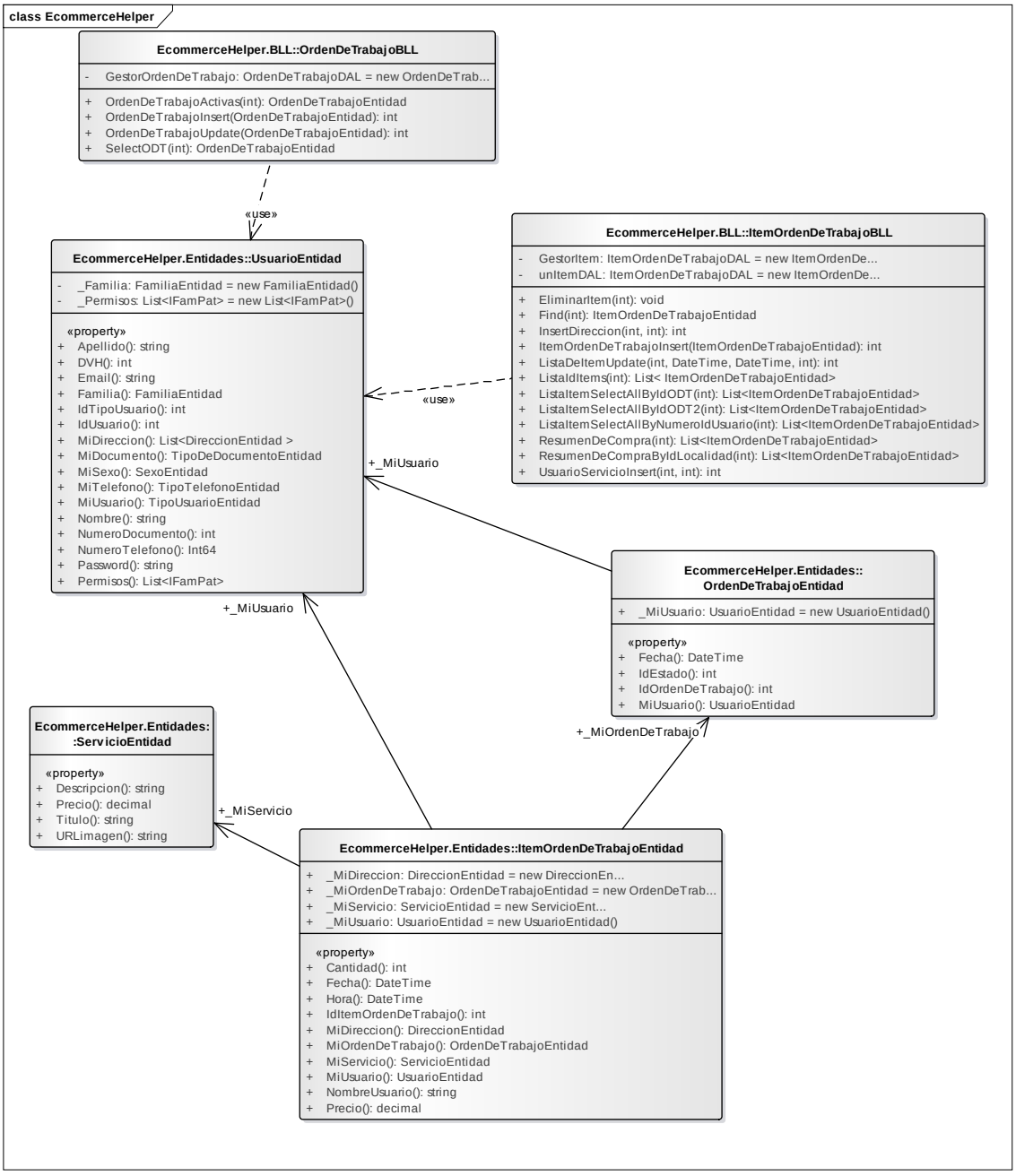
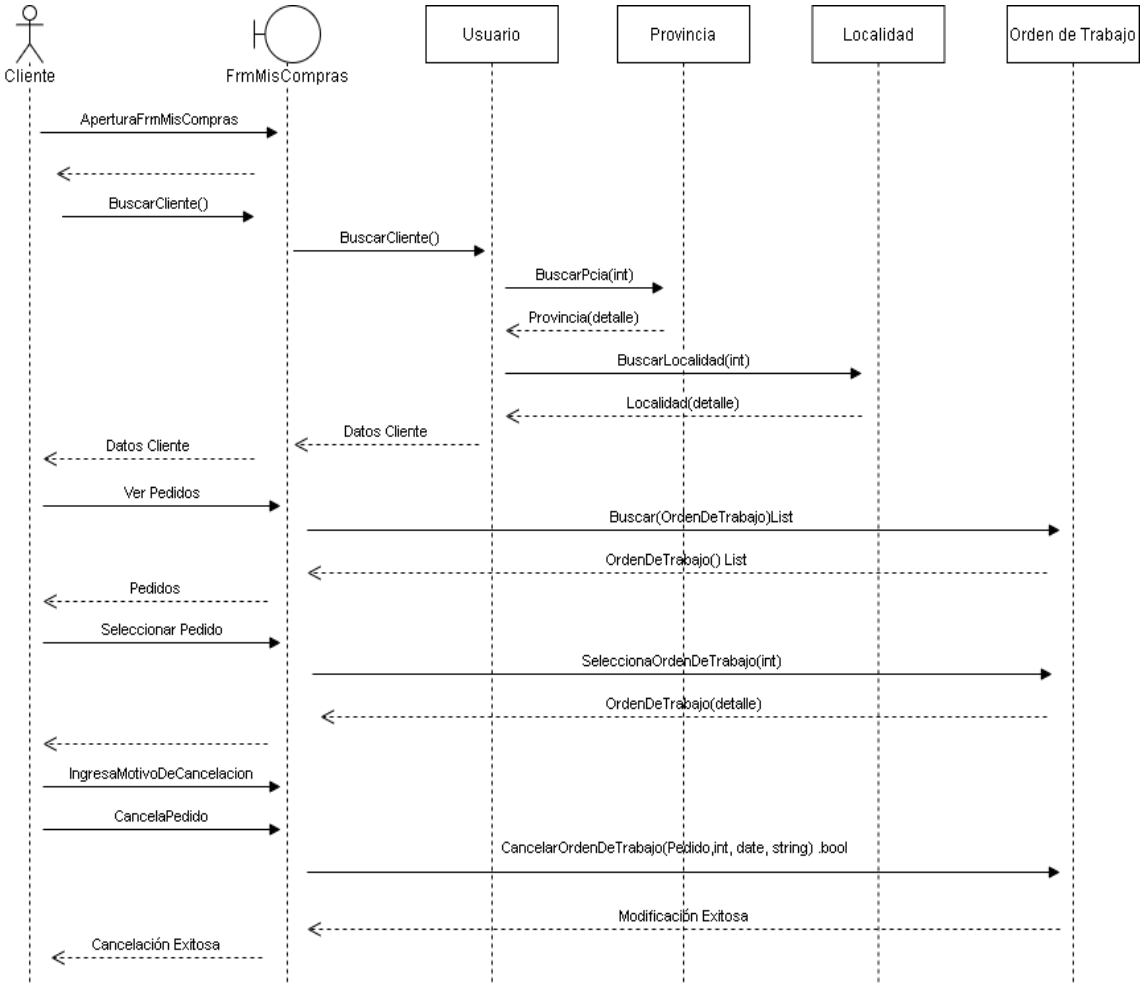

class EcommerceHelper
EcommerceHelper.BLL::OrdenDeTrabajoBLL
- GestorOrdenDeTrabajo: OrdenDeTrabajoDAL = new OrdenDeTrab...
+ OrdenDeTrabajoActivas(int): OrdenDeTrabajoEntidad
+ OrdenDeTrabajoInsert(OrdenDeTrabajoEntidad): int
+ OrdenDeTrabajoUpdate(OrdenDeTrabajoEntidad): int
+ SelectODT(int): OrdenDeTrabajoEntidad
«use»
EcommerceHelper.Entidades::UsuarioEntidad
- _Familia: FamiliaEntidad = new FamiliaEntidad()
- _Permisos: List<IFamPat> = new List<IFamPat>()
«property»
+ Apellido(): string
+ DVH(): int
+ Email(): string
+ Familia(): FamiliaEntidad
+ IdTipoUsuario(): int
+ IdUsuario(): int
+ MiDireccion(): List<DireccionEntidad >
+ MiDocumento(): TipoDeDocumentoEntidad
+ MiSexo(): SexoEntidad
+ MiTelefono(): TipoTelefonoEntidad
+ MiUsuario(): TipoUsuarioEntidad
+ Nombre(): string
+ NumeroDocumento(): int
+ NumeroTelefono(): Int64
+ Password(): string
+ Permisos(): List<IFamPat>
EcommerceHelper.BLL::ItemOrdenDeTrabajoBLL
- GestorItem: ItemOrdenDeTrabajoDAL = new ItemOrdenDe...
- unItemDAL: ItemOrdenDeTrabajoDAL = new ItemOrdenDe...
+ EliminarItem(int): void
+ Find(int): ItemOrdenDeTrabajoEntidad
+ InsertDireccion(int, int): int
+ ItemOrdenDeTrabajoInsert(ItemOrdenDeTrabajoEntidad): int
+ ListaDeItemUpdate(int, DateTime, DateTime, int): int
+ ListaIdItems(int): List< ItemOrdenDeTrabajoEntidad>
+ ListaItemSelectAllByIdODT(int): List<ItemOrdenDeTrabajoEntidad>
+ ListaItemSelectAllByIdODT2(int): List<ItemOrdenDeTrabajoEntidad>
+ ListaItemSelectAllByNumeroIdUsuario(int): List<ItemOrdenDeTrabajoEntidad>
+ ResumenDeCompra(int): List<ItemOrdenDeTrabajoEntidad>
+ ResumenDeCompraByIdLocalidad(int): List<ItemOrdenDeTrabajoEntidad>
+ UsuarioServicioInsert(int, int): int
«use»
+_MiUsuario
+_MiUsuario
EcommerceHelper.Entidades::
OrdenDeTrabajoEntidad
+ _MiUsuario: UsuarioEntidad = new UsuarioEntidad()
«property»
+ Fecha(): DateTime
+ IdEstado(): int
+ IdOrdenDeTrabajo(): int
+ MiUsuario(): UsuarioEntidad
+_MiOrdenDeTrabajo
EcommerceHelper.Entidades:
:ServicioEntidad
«property»
+ Descripcion(): string
+ Precio(): decimal
+ Titulo(): string
+ URLimagen(): string
+_MiServicio
EcommerceHelper.Entidades::ItemOrdenDeTrabajoEntidad
+ _MiDireccion: DireccionEntidad = new DireccionEn...
+ _MiOrdenDeTrabajo: OrdenDeTrabajoEntidad = new OrdenDeTrab...
+ _MiServicio: ServicioEntidad = new ServicioEnt...
+ _MiUsuario: UsuarioEntidad = new UsuarioEntidad()
«property»
+ Cantidad(): int
+ Fecha(): DateTime
+ Hora(): DateTime
+ IdItemOrdenDeTrabajo(): int
+ MiDireccion(): DireccionEntidad
+ MiOrdenDeTrabajo(): OrdenDeTrabajoEntidad
+ MiServicio(): ServicioEntidad
+ MiUsuario(): UsuarioEntidad
+ NombreUsuario(): string
+ Precio(): decimal

Diagrama de Secuencia CU - 006 – Cancelación de Venta



Pantalla

Titulo	Descripcion	Precio	Imagen	Fecha	Hora	Cantidad	
Limpieza Exterior	Quinchos, Jardines de invierno. Explanadas y veredas. Galerias, Glorietas. Galerias y mas!! 4 horas	1000		30/11/2020	08:00	1	Eliminar

CU - 027 – Alta Servicio

Precondición
El usuario debe estar logueado con los permisos correspondientes para poder acceder al alta de servicio, Administrador.

Actores primarios:
Administrador

Camino principal del caso de uso – Alta Servicio

1. El usuario cliente selecciona crear un nuevo servicio.
2. El sistema solicita se ingrese:
    a. Titulo.
    b. Descripción
    c. Imagen.
    d. Precio unitario.

3. El usuario cliente ingresa los datos solicitados y presiona el botón Nuevo confirmando la operación alta servicio.
4. El sistema valida los datos.
5. El sistema da de alta el servicio en el sistema.
6. El sistema registra en la bitácora el evento de la creación del servicio.
7. El sistema refresca la lista de servicio y lo muestra en la grilla.

Caminos alternativos

Para el paso 4: el sistema detecta que falta ingresar información o la misma no es del tipo indicado

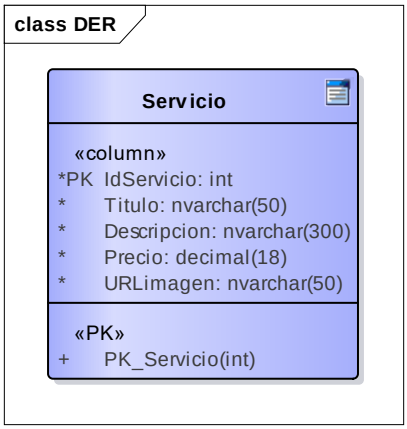
4.1 El sistema informa que aún falta ingresar datos o los datos ingresados no son del tipo indicado.

Flujo alternativo

Falla en el momento de grabación del alta de servicio.
El sistema registra en bitácora el error producido.

Postcondición
El servicio se encuentra creado.

uc Caso de Uso
Usuario Empleado
CU - 027 – Alta de Servicio


class DER
Servicio
«column»
*PK IdServicio: int
* Titulo: nvarchar(50)
* Descripcion: nvarchar(300)
* Precio: decimal(18)
* URLimagen: nvarchar(50)
«PK»
+ PK_Servicio(int)

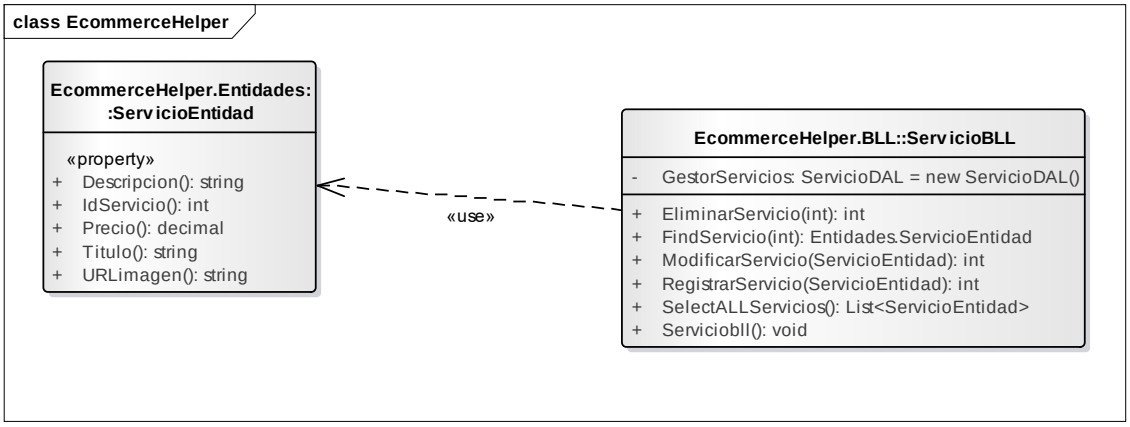
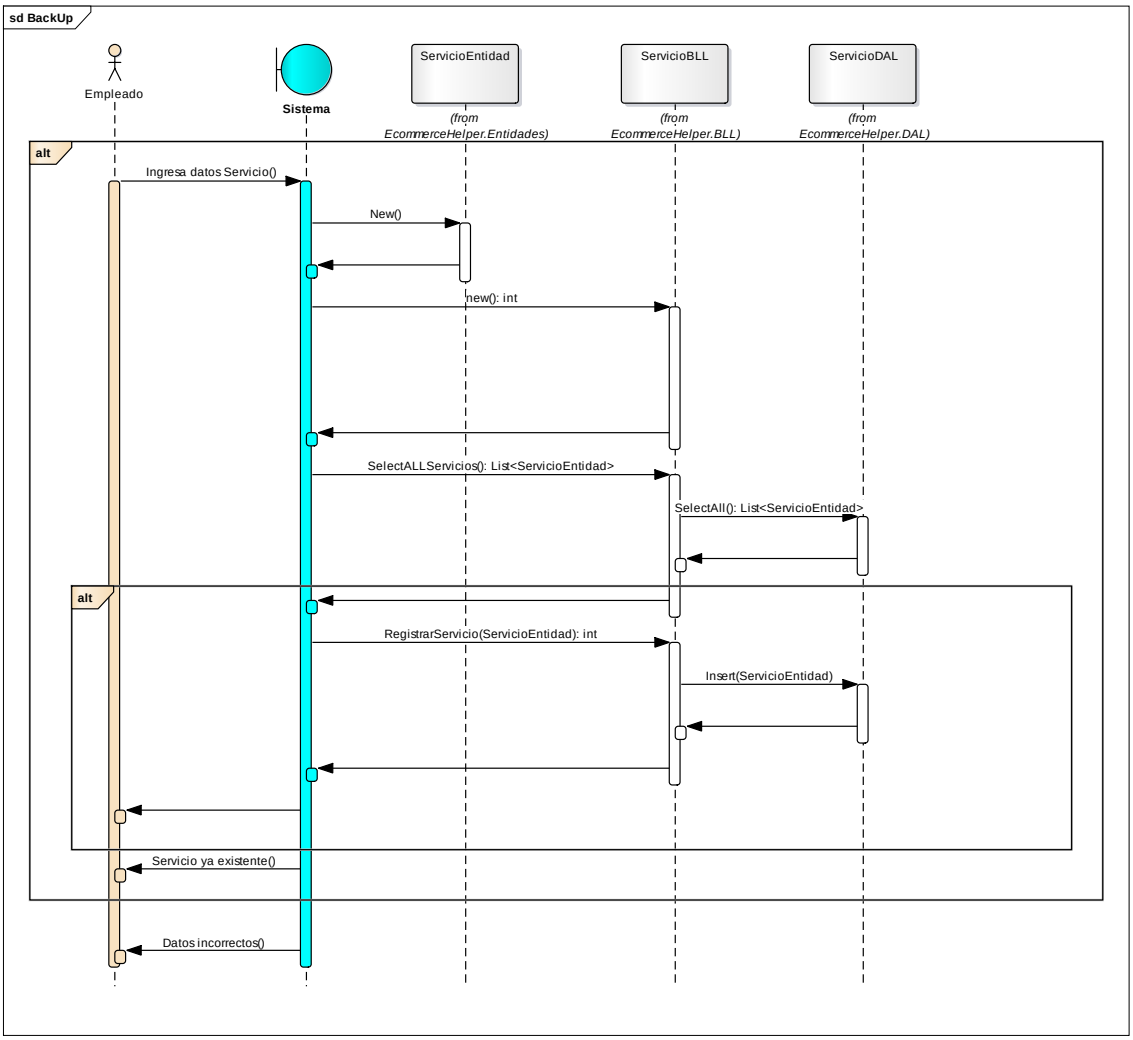
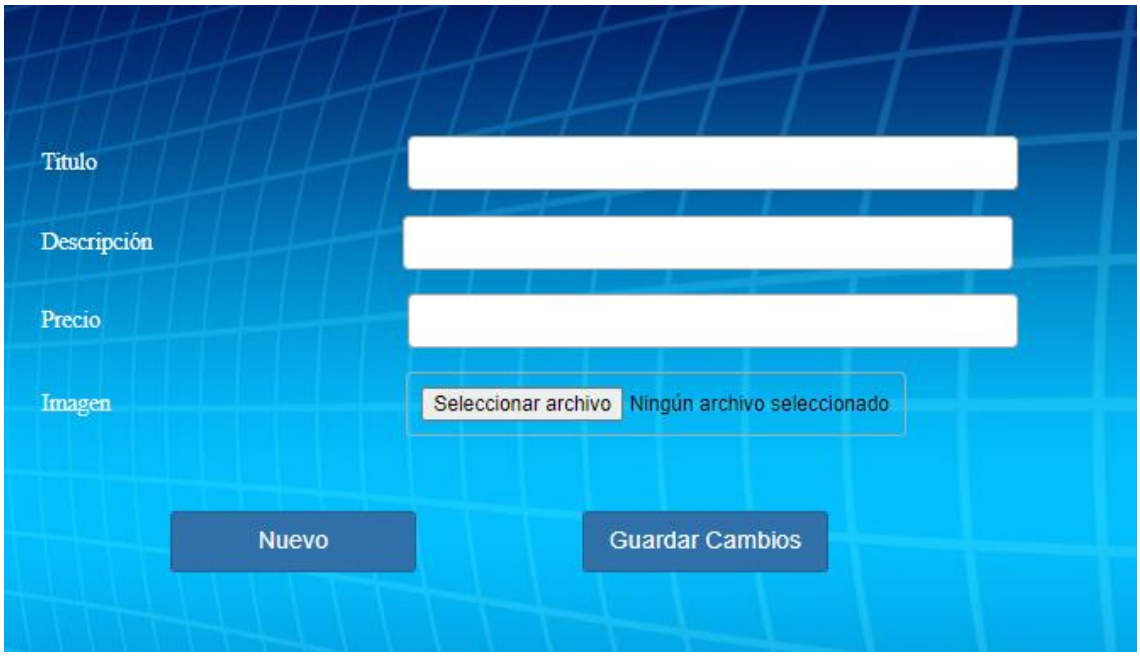

class EcommerceHelper
EcommerceHelper.Entidades::ServicioEntidad
«property»
+ Descripcion(): string
+ IdServicio(): int
+ Precio(): decimal
+ Titulo(): string
+ URLimagen(): string
«use»
EcommerceHelper.BLL::ServicioBLL
- GestorServicios: ServicioDAL = new ServicioDAL()
+ EliminarServicio(int): int
+ FindServicio(int): Entidades.ServicioEntidad
+ ModificarServicio(ServicioEntidad): int
+ RegistrarServicio(ServicioEntidad): int
+ SelectALLServicios(): List<ServicioEntidad>
+ ServiciobIl(): void

Diagrama de Secuencia CU - 027 – Alta de Servicio



Pantalla


Titulo
Descripción
Precio
Imagen
Seleccionar archivo
Ningún archivo seleccionado
Nuevo
Guardar Cambios

CU - 028 – Modificar Servicio

Precondición
El usuario debe estar logueado con los permisos correspondientes para poder acceder al modificar el servicio. Administrador

Actores primarios:
Administrador

Camino principal del caso de uso – Modificar Servicio

1. El usuario cliente selecciona de la grilla de servicios el que desea modificar.
2. El sistema muestra los datos actuales del servicio en el formulario los siguientes datos.
    a. Titulo.
    b. Descripción
    c. Precio unitario.

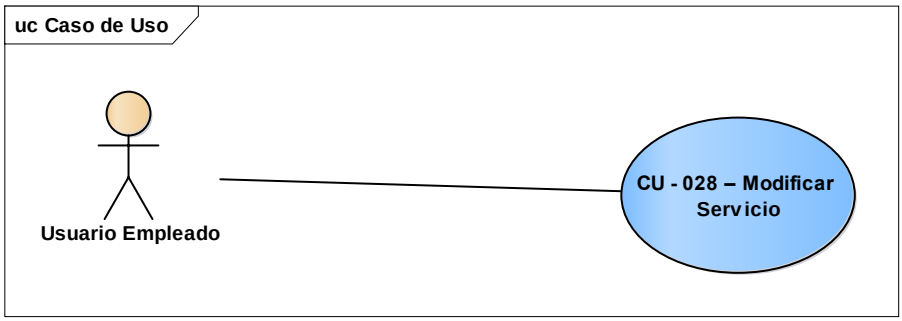
3. El sistema habilito para que el usuario pueda seleccionar una nueva imagen.
4. El usuario ingresa los datos a ser modificados y confirma la operación presionando el botón Guardar Cambios.
5. El sistema valida los datos.
6. El sistema realiza las modificaciones introducidas por el usuario del servicio en el sistema.
7. El sistema refresca la lista de servicio y lo muestra en la grilla.

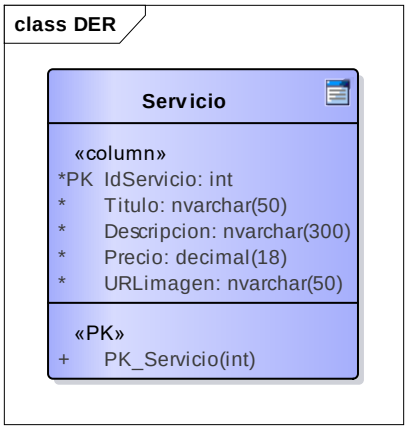
Caminos alternativos

Para el paso 5: el sistema detecta que falta ingresar información o la misma no es del tipo indicado

5.1 El sistema informa que aún falta ingresar datos o los datos ingresados no son del tipo indicado.

Postcondición
El servicio se encuentra modificado.


uc Caso de Uso
Usuario Empleado
CU - 028 – Modificar Servicio


class DER
Servicio
«column»
*PK IdServicio: int
* Titulo: nvarchar(50)
* Descripcion: nvarchar(300)
* Precio: decimal(18)
* URLimagen: nvarchar(50)
«PK»
+ PK_Servicio(int)

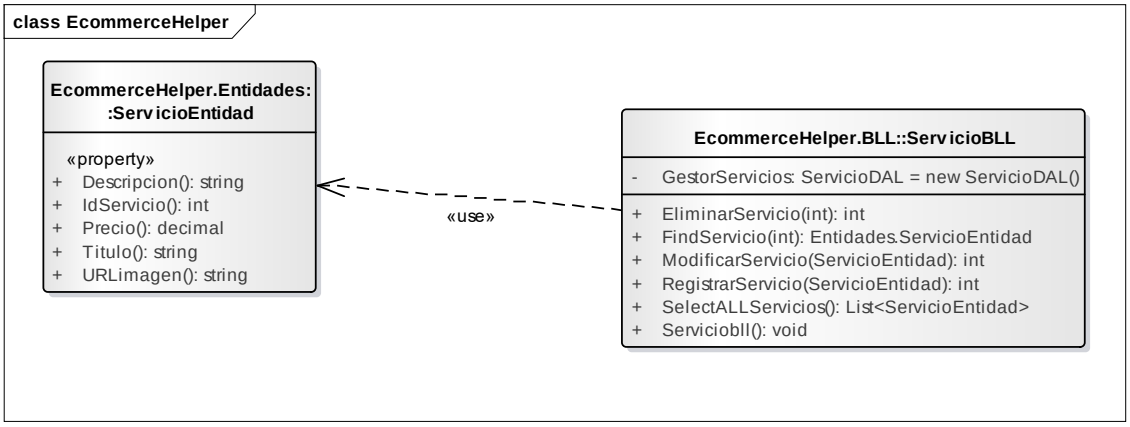
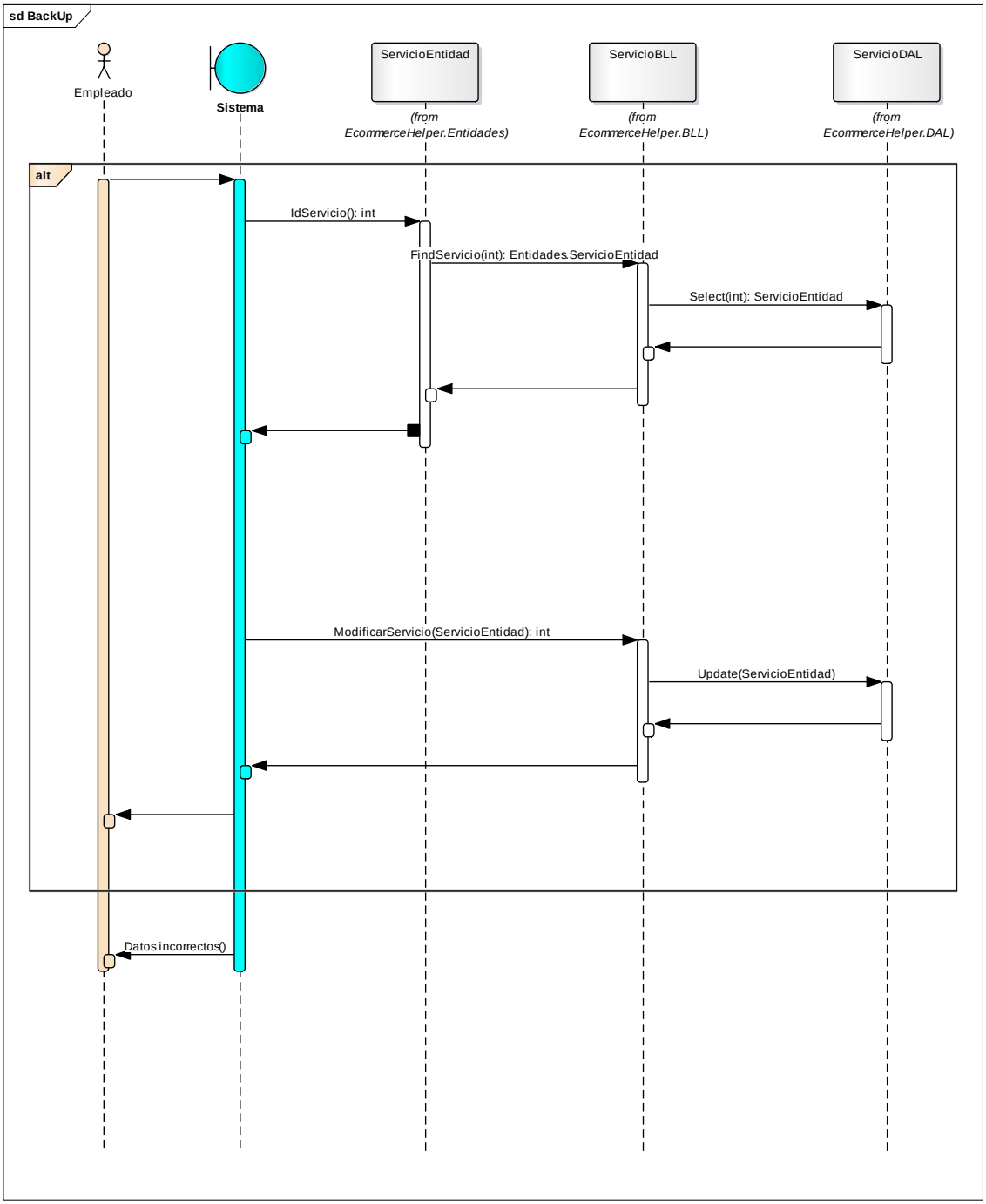
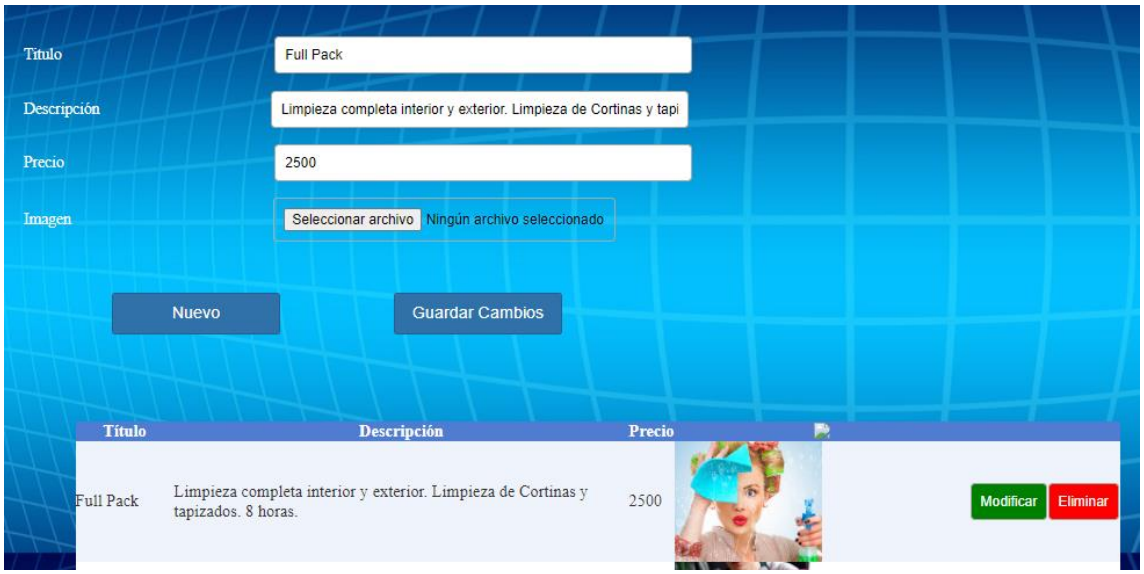

class EcommerceHelper
EcommerceHelper.Entidades::ServicioEntidad
«property»
+ Descripcion(): string
+ IdServicio(): int
+ Precio(): decimal
+ Titulo(): string
+ URLimagen(): string
«use»
EcommerceHelper.BLL::ServicioBLL
- GestorServicios: ServicioDAL = new ServicioDAL()
+ EliminarServicio(int): int
+ FindServicio(int): Entidades.ServicioEntidad
+ ModificarServicio(ServicioEntidad): int
+ RegistrarServicio(ServicioEntidad): int
+ SelectALLServicios(): List<ServicioEntidad>
+ ServiciobIl(): void

Diagrama de Secuencia CU - 028 – Modificar Servicio



Pantalla


Título
Full Pack
Descripción
Limpieza completa interior y exterior. Limpieza de Cortinas y tapi
Precio
2500
Imagen
Seleccionar archivo
Ningún archivo seleccionado
Nuevo
Guardar Cambios
Título
Descripción
Precio
Full Pack
Limpieza completa interior y exterior. Limpieza de Cortinas y tapizados. 8 horas.
2500
Modificar
Eliminar

CU - 029 – Eliminar Servicio

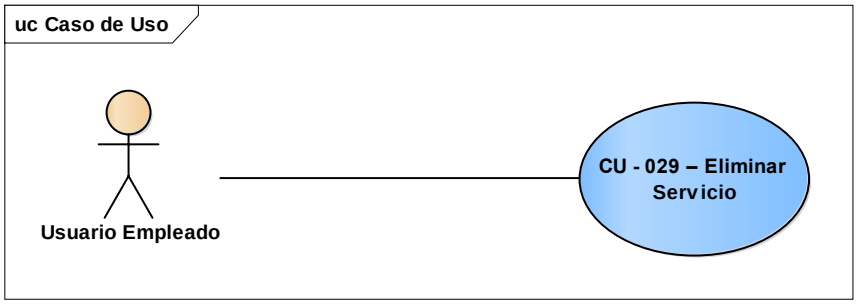
Precondición
El usuario debe estar logueado con los permisos correspondientes para poder acceder a eliminar el servicio. Administrador

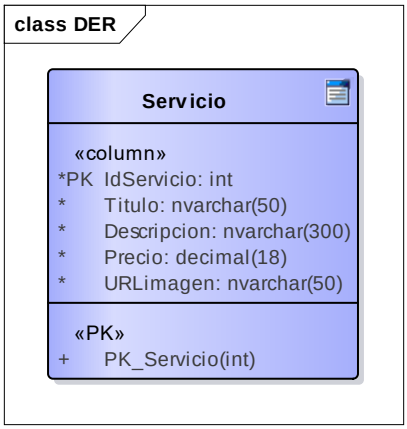
Actores primarios:
Administrador

Camino principal del caso de uso – Eliminar Servicio

1. El usuario cliente selecciona de la grilla de servicios el que desea eliminar.
2. El sistema elimina el servicio.
3. El sistema refresca la grilla y actualiza los servicios disponibles.

Postcondición
El servicio se encuentra eliminado.


uc Caso de Uso
Usuario Empleado
CU - 029 – Eliminar Servicio


class DER
Servicio
«column»
*PK IdServicio: int
* Titulo: nvarchar(50)
* Descripcion: nvarchar(300)
* Precio: decimal(18)
* URLimagen: nvarchar(50)
«PK»
+ PK_Servicio(int)

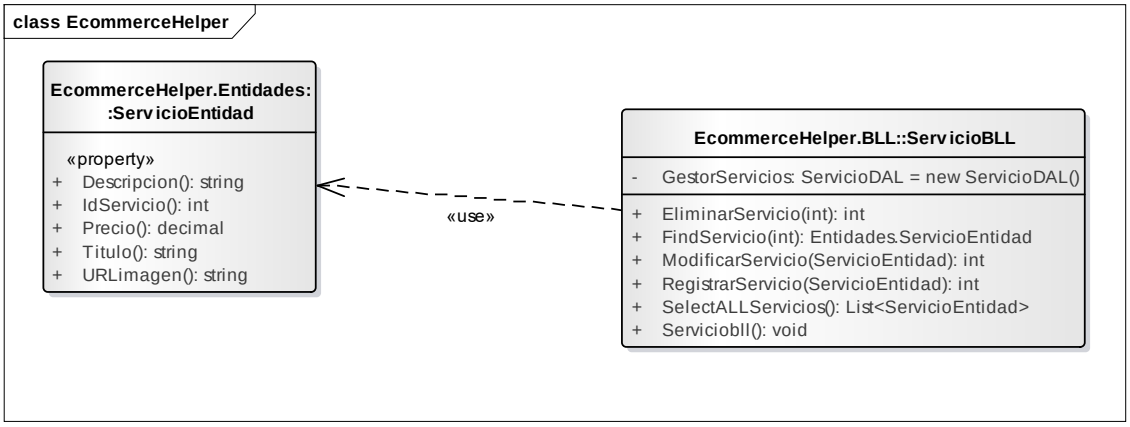
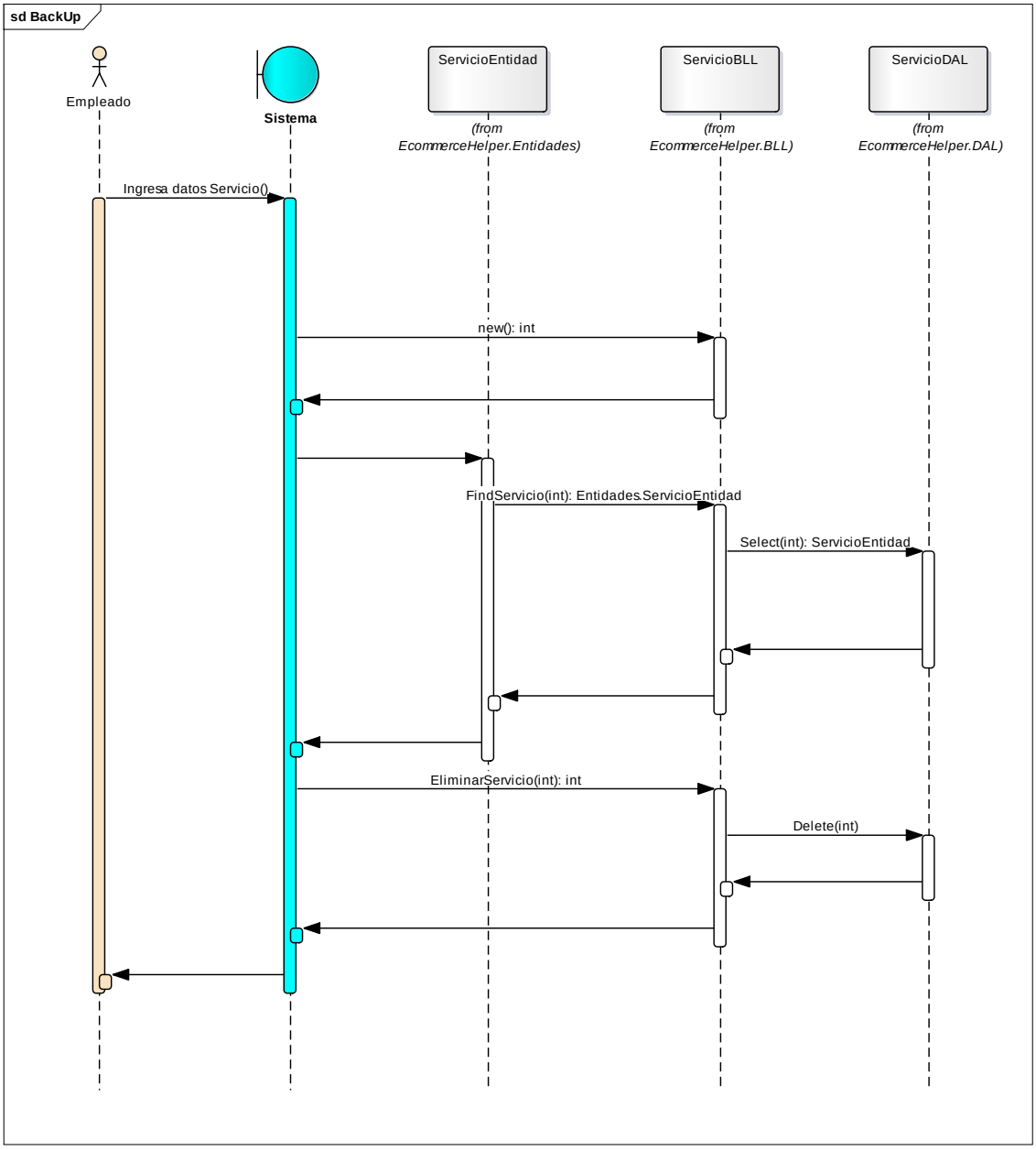

class EcommerceHelper
EcommerceHelper.Entidades::ServicioEntidad
«property»
+ Descripcion(): string
+ IdServicio(): int
+ Precio(): decimal
+ Titulo(): string
+ URLimagen(): string
«use»
EcommerceHelper.BLL::ServicioBLL
- GestorServicios: ServicioDAL = new ServicioDAL()
+ EliminarServicio(int): int
+ FindServicio(int): Entidades.ServicioEntidad
+ ModificarServicio(ServicioEntidad): int
+ RegistrarServicio(ServicioEntidad): int
+ SelectALLServicios(): List<ServicioEntidad>
+ ServiciobIl(): void

Diagrama de Secuencia CU - 029 – Eliminar Servicio



Pantalla

Título	Descripción	Precio		
Full Pack	Limpieza completa interior y exterior. Limpieza de Cortinas y tapizados. 8 horas.	2500		Modificar Eliminar
Limpieza de Tapizados	Limpieza de sillas y sillones en Cuero , cuerinas, panas y telas. Hasta 2 m²	500		Modificar Eliminar
Limpieza de Cortinas	Limpia tu cortina Black-out . Cortinas de Tela Tupida. Estilos Señoriales y Bando. Enrollables y mas!	500		Modificar Eliminar
Limpieza Interior	Limpieza dormitorios. Limpieza baños, living, cocinas, playroom y mas. Consulta planes a tu medida! 6 horas	1500		Modificar Eliminar
Limpieza Exterior	Quinchos, Jardines de invierno. Explanadas y veredas. Galerias, Glorietas. Galerias y mas!! 4 horas	1000		Modificar Eliminar

CU - 032 – Alta dirección

Precondición
El cliente debe estar logueado.

Actores primarios:
Usuario cliente

Camino principal del caso de uso – Alta dirección

1. El usuario cliente selecciona agregar dirección.
2. El sistema despliega el formulario para agregar una nueva dirección.
3. El sistema solicita se ingrese:

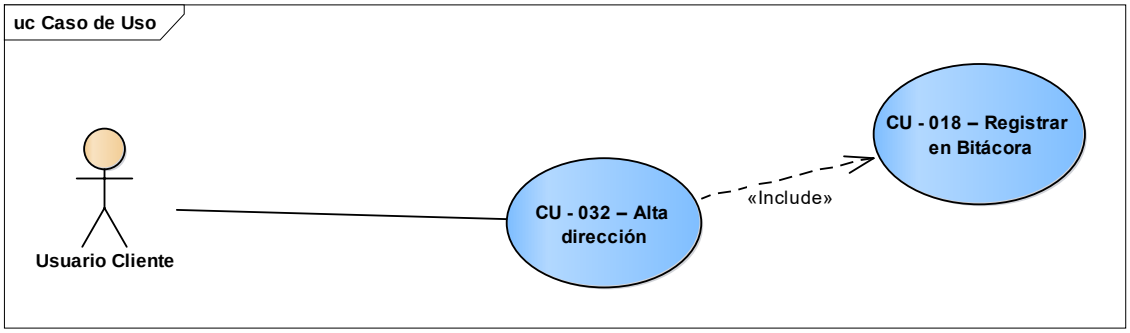
    i. Provincia: texto
    ii. Localidad: texto
    iii. Calle: texto
    iv. Nro. de calle: Números
    v. Piso: Número.
    vi. Departamento: texto
    vii. Tipo de Domicilio

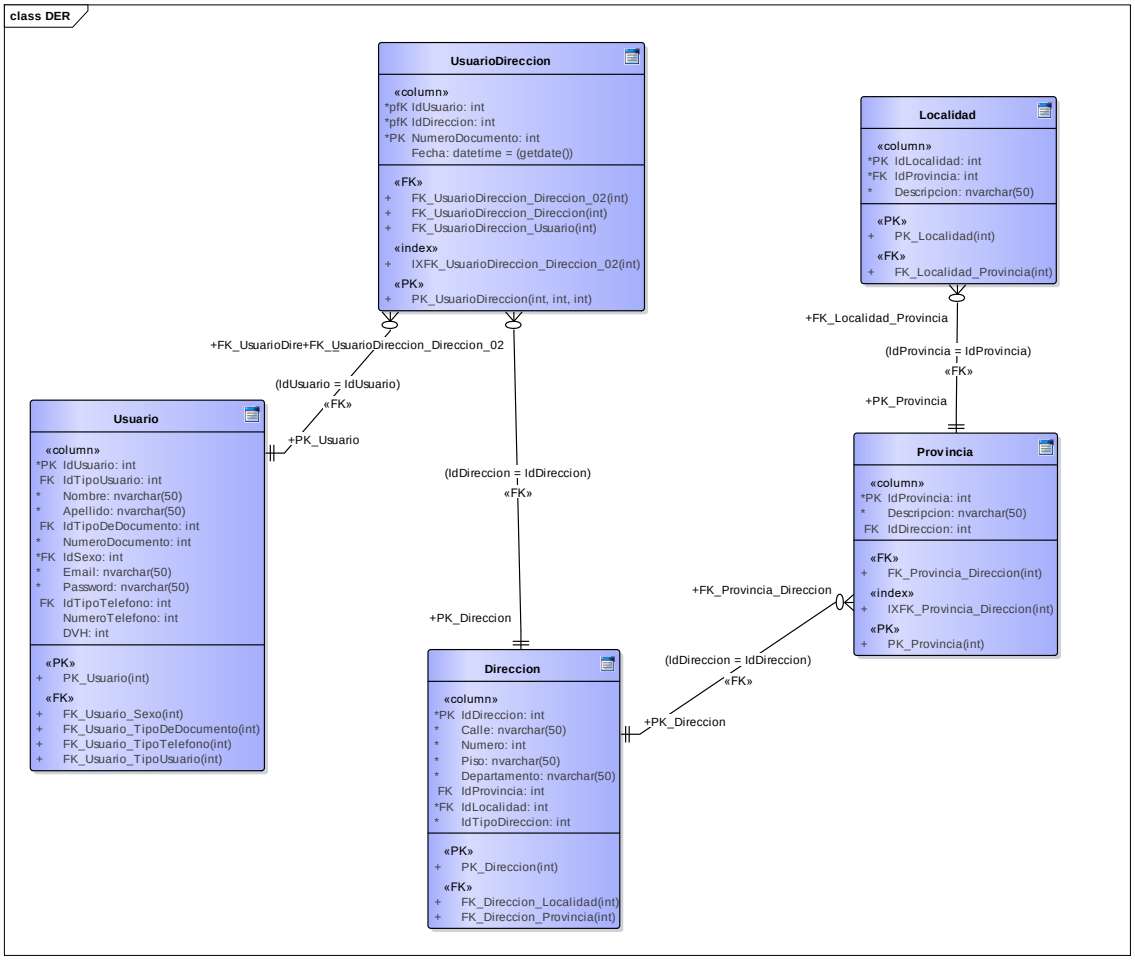
4. El usuario cliente ingresa los datos solicitados y confirma la operación agregar una nueva dirección.
5. El sistema valida los datos.
6. El sistema registra la nueva dirección del usuario cliente.

Postcondición
El cliente ha registrado una nueva dirección.

Flujo Alternativo

El cliente presiona el botón cancelar y retorna a la grilla de “mis direcciones”.


uc Caso de Uso
Usuario Cliente
CU - 032 – Alta dirección
«Include»
CU - 018 – Registrar en Bitácora


class DER
UsuarioDireccion
«column»
*pfK IdUsuario: int
*pfK IdDireccion: int
*PK NumeroDocumento: int
Fecha: datetime = (getdate())
«FK»
+ FK_UsuarioDireccion_Direccion_02(int)
+ FK_UsuarioDireccion_Direccion(int)
+ FK_UsuarioDireccion_Usuario(int)
«index»
+ IXFK_UsuarioDireccion_Direccion_02(int)
«PK»
+ PK_UsuarioDireccion(int, int, int)
Localidad
«column»
*PK IdLocalidad: int
*FK IdProvincia: int
* Descripcion: nvarchar(50)
«PK»
+ PK_Localidad(int)
«FK»
+ FK_Localidad_Provincia(int)
+FK_Localidad_Provincia
(IdProvincia = IdProvincia)
«FK»
+PK_Provincia
+FK_UsuarioDire+FK_UsuarioDireccion_Direccion_02
(IdUsuario = IdUsuario)
«FK»
+PK_Usuario
Usuario
«column»
*PK IdUsuario: int
FK IdTipoUsuario: int
* Nombre: nvarchar(50)
* Apellido: nvarchar(50)
FK IdTipoDeDocumento: int
* NumeroDocumento: int
*FK IdSexo: int
* Email: nvarchar(50)
* Password: nvarchar(50)
FK IdTipoTelefono: int
NumeroTelefono: int
DVH: int
«PK»
+ PK_Usuario(int)
«FK»
+ FK_Usuario_Sexo(int)
+ FK_Usuario_TipoDeDocumento(int)
+ FK_Usuario_TipoTelefono(int)
+ FK_Usuario_TipoUsuario(int)
(IdDireccion = IdDireccion)
«FK»
+PK_Direccion
Provincia
«column»
*PK IdProvincia: int
* Descripcion: nvarchar(50)
FK IdDireccion: int
«FK»
+ FK_Provincia_Direccion(int)
«index»
+ IXFK_Provincia_Direccion(int)
«PK»
+ PK_Provincia(int)
+FK_Provincia_Direccion
(IdDireccion = IdDireccion)
«FK»
+PK_Direccion
Direccion
«column»
*PK IdDireccion: int
* Calle: nvarchar(50)
* Numero: int
* Piso: nvarchar(50)
* Departamento: nvarchar(50)
FK IdProvincia: int
*FK IdLocalidad: int
* IdTipoDireccion: int
«PK»
+ PK_Direccion(int)
«FK»
+ FK_Direccion_Localidad(int)
+ FK_Direccion_Provincia(int)

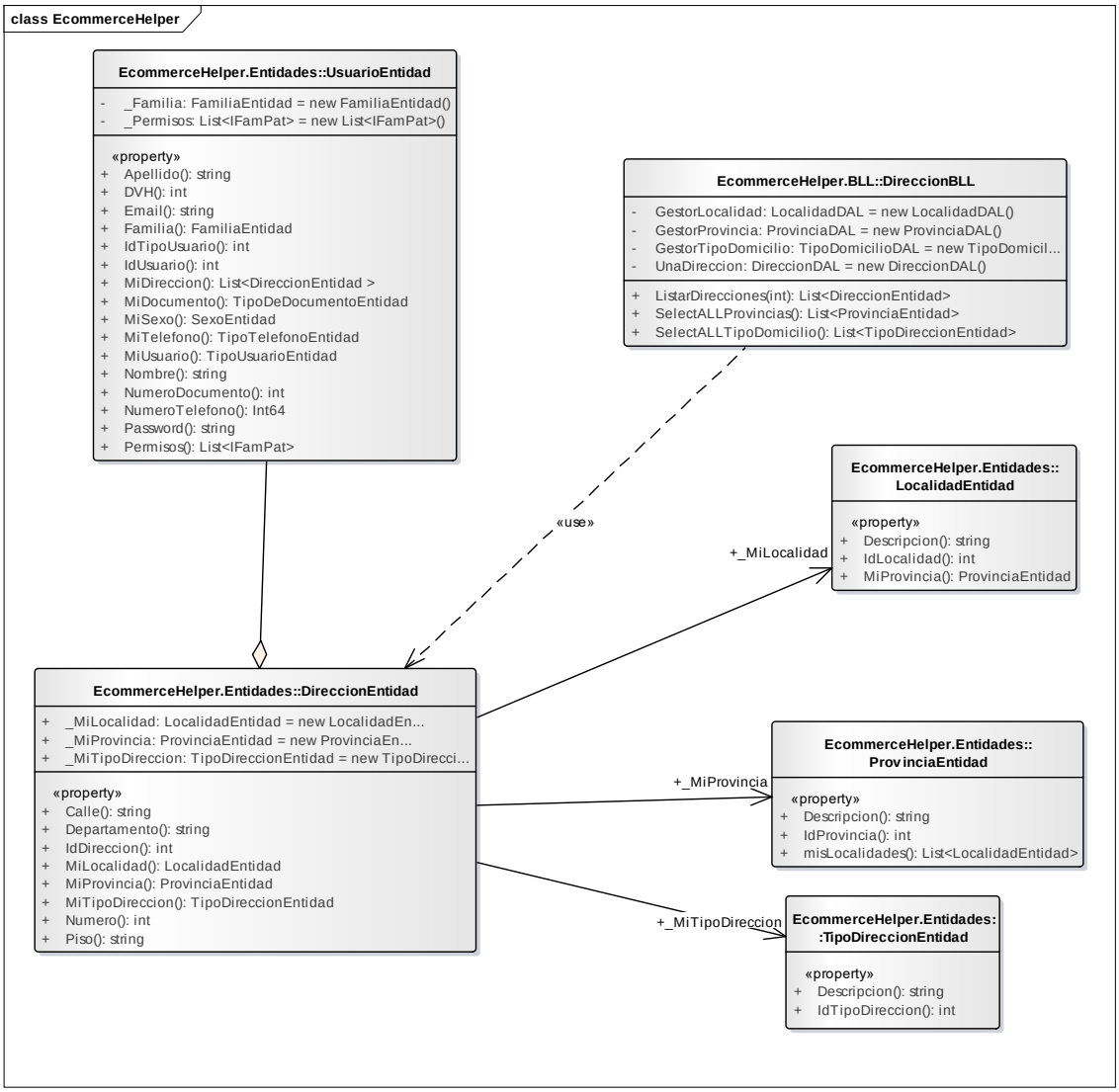
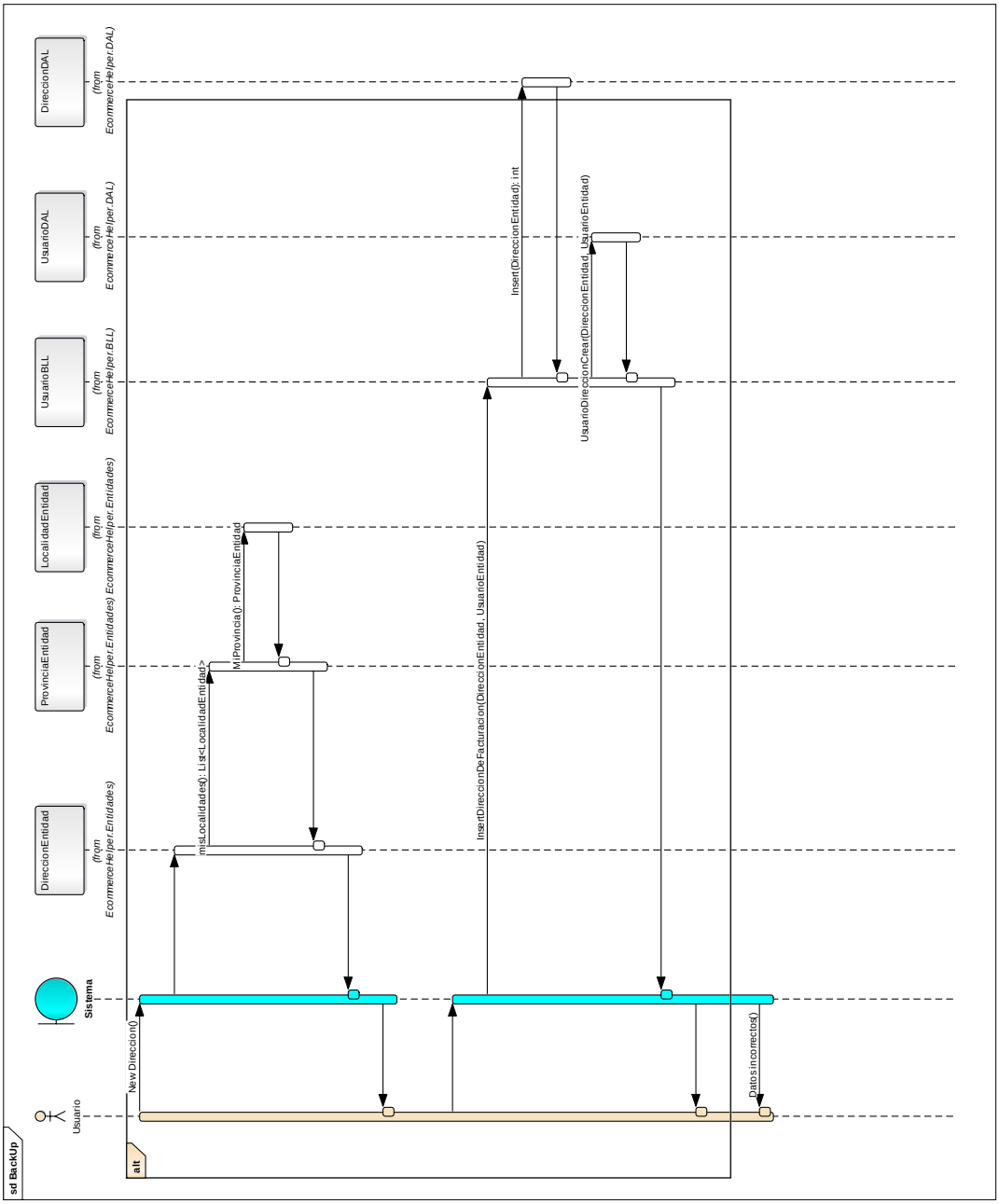
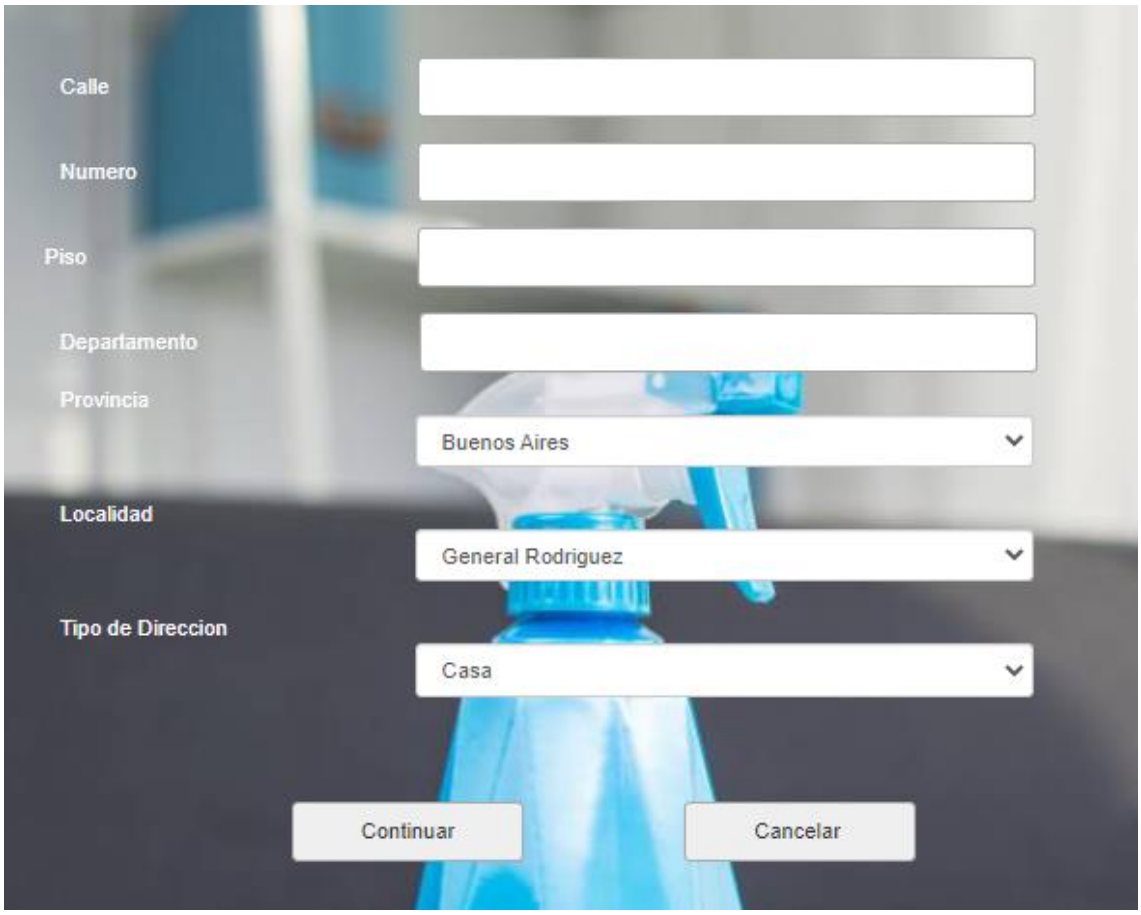

class EcommerceHelper
EcommerceHelper.Entidades::UsuarioEntidad
- _Familia: FamiliaEntidad = new FamiliaEntidad()
- _Permisos: List<IFamPat> = new List<IFamPat>()
«property»
+ Apellido(): string
+ DVH(): int
+ Email(): string
+ Familia(): FamiliaEntidad
+ IdTipoUsuario(): int
+ IdUsuario(): int
+ MiDireccion(): List<DireccionEntidad >
+ MiDocumento(): TipoDeDocumentoEntidad
+ MiSexo(): SexoEntidad
+ MiTelefono(): TipoTelefonoEntidad
+ MiUsuario(): TipoUsuarioEntidad
+ Nombre(): string
+ NumeroDocumento(): int
+ NumeroTelefono(): Int64
+ Password(): string
+ Permisos(): List<IFamPat>
EcommerceHelper.BLL::DireccionBLL
- GestorLocalidad: LocalidadDAL = new LocalidadDAL()
- GestorProvincia: ProvinciaDAL = new ProvinciaDAL()
- GestorTipoDomicilio: TipoDomicilioDAL = new TipoDomicil...
- UnaDireccion: DireccionDAL = new DireccionDAL()
+ ListarDirecciones(int): List<DireccionEntidad>
+ SelectALLProvincias(): List<ProvinciaEntidad>
+ SelectALLTipoDomicilio(): List<TipoDireccionEntidad>
«use»
EcommerceHelper.Entidades::LocalidadEntidad
«property»
+ Descripcion(): string
+ IdLocalidad(): int
+ MiProvincia(): ProvinciaEntidad
+_MiLocalidad
EcommerceHelper.Entidades::DireccionEntidad
+ _MiLocalidad: LocalidadEntidad = new LocalidadEn...
+ _MiProvincia: ProvinciaEntidad = new ProvinciaEn...
+ _MiTipoDireccion: TipoDireccionEntidad = new TipoDirecci...
«property»
+ Calle(): string
+ Departamento(): string
+ IdDireccion(): int
+ MiLocalidad(): LocalidadEntidad
+ MiProvincia(): ProvinciaEntidad
+ MiTipoDireccion(): TipoDireccionEntidad
+ Numero(): int
+ Piso(): string
+_MiProvincia
EcommerceHelper.Entidades::ProvinciaEntidad
«property»
+ Descripcion(): string
+ IdProvincia(): int
+ misLocalidades(): List<LocalidadEntidad>
+_MiTipoDireccion
EcommerceHelper.Entidades::TipoDireccionEntidad
«property»
+ Descripcion(): string
+ IdTipoDireccion(): int

Diagrama de Secuencia CU - 032 – Alta dirección



Pantalla


Calle
Numero
Piso
Departamento
Provincia
Buenos Aires
Localidad
General Rodriguez
Tipo de Direccion
Casa
Continuar
Cancelar

CU - 033 – Eliminar dirección

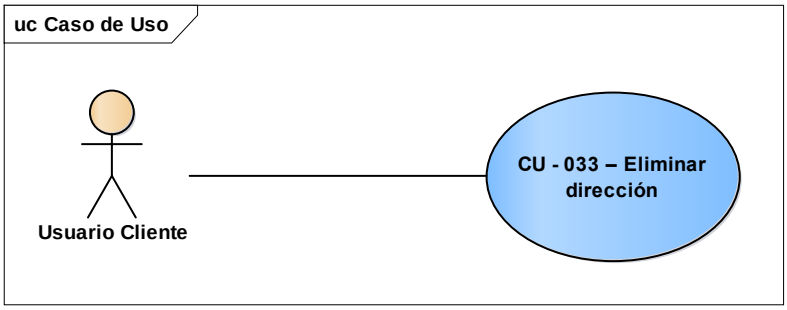
Precondición
El cliente debe estar logueado.

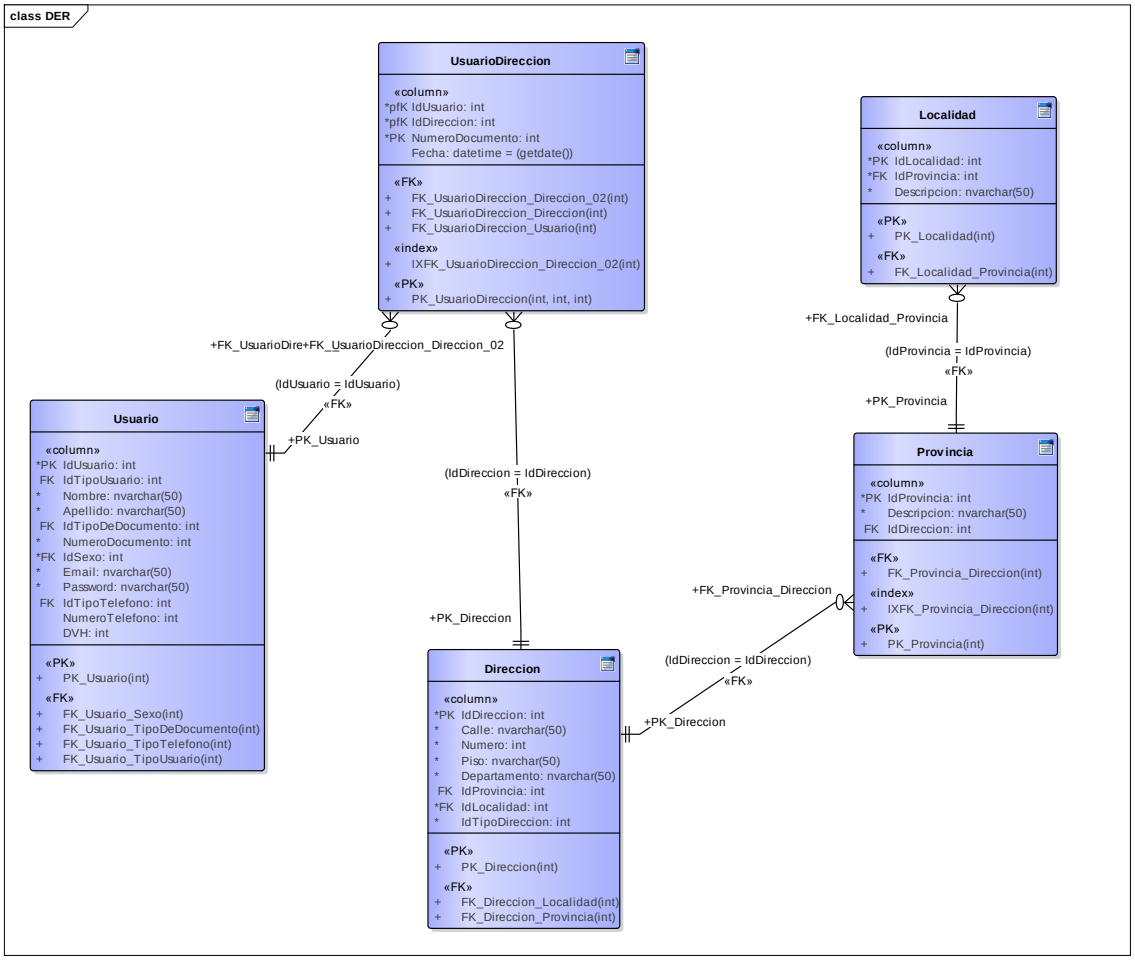
Actores primarios:
Usuario cliente

Camino principal del caso de uso – Eliminar dirección

1. El usuario cliente selecciona de la grilla de direcciones la que desea eliminar.
2. El sistema elimina la dirección.
3. El sistema refresca la grilla y actualiza las direcciones disponibles.

Postcondición
El cliente ha eliminado una dirección.


uc Caso de Uso
Usuario Cliente
CU - 033 – Eliminar dirección


class DER
UsuarioDireccion
«column»
*pfK IdUsuario: int
*pfK IdDireccion: int
*PK NumeroDocumento: int
Fecha: datetime = (getdate())
«FK»
+ FK_UsuarioDireccion_Direccion_02(int)
+ FK_UsuarioDireccion_Direccion(int)
+ FK_UsuarioDireccion_Usuario(int)
«index»
+ IXFK_UsuarioDireccion_Direccion_02(int)
«PK»
+ PK_UsuarioDireccion(int, int, int)
Localidad
«column»
*PK IdLocalidad: int
*FK IdProvincia: int
* Descripcion: nvarchar(50)
«PK»
+ PK_Localidad(int)
«FK»
+ FK_Localidad_Provincia(int)
+FK_Localidad_Provincia
(IdProvincia = IdProvincia)
«FK»
+PK_Provincia
+FK_UsuarioDire+FK_UsuarioDireccion_Direccion_02
(IdUsuario = IdUsuario)
«FK»
+PK_Usuario
Usuario
«column»
*PK IdUsuario: int
FK IdTipoUsuario: int
* Nombre: nvarchar(50)
* Apellido: nvarchar(50)
FK IdTipoDeDocumento: int
* NumeroDocumento: int
*FK IdSexo: int
* Email: nvarchar(50)
* Password: nvarchar(50)
FK IdTipoTelefono: int
NumeroTelefono: int
DVH: int
«PK»
+ PK_Usuario(int)
«FK»
+ FK_Usuario_Sexo(int)
+ FK_Usuario_TipoDeDocumento(int)
+ FK_Usuario_TipoTelefono(int)
+ FK_Usuario_TipoUsuario(int)
(IdDireccion = IdDireccion)
«FK»
+PK_Direccion
Provincia
«column»
*PK IdProvincia: int
* Descripcion: nvarchar(50)
FK IdDireccion: int
«FK»
+ FK_Provincia_Direccion(int)
«index»
+ IXFK_Provincia_Direccion(int)
«PK»
+ PK_Provincia(int)
+FK_Provincia_Direccion
(IdDireccion = IdDireccion)
«FK»
+PK_Direccion
Direccion
«column»
*PK IdDireccion: int
* Calle: nvarchar(50)
* Numero: int
* Piso: nvarchar(50)
* Departamento: nvarchar(50)
FK IdProvincia: int
*FK IdLocalidad: int
* IdTipoDireccion: int
«PK»
+ PK_Direccion(int)
«FK»
+ FK_Direccion_Localidad(int)
+ FK_Direccion_Provincia(int)

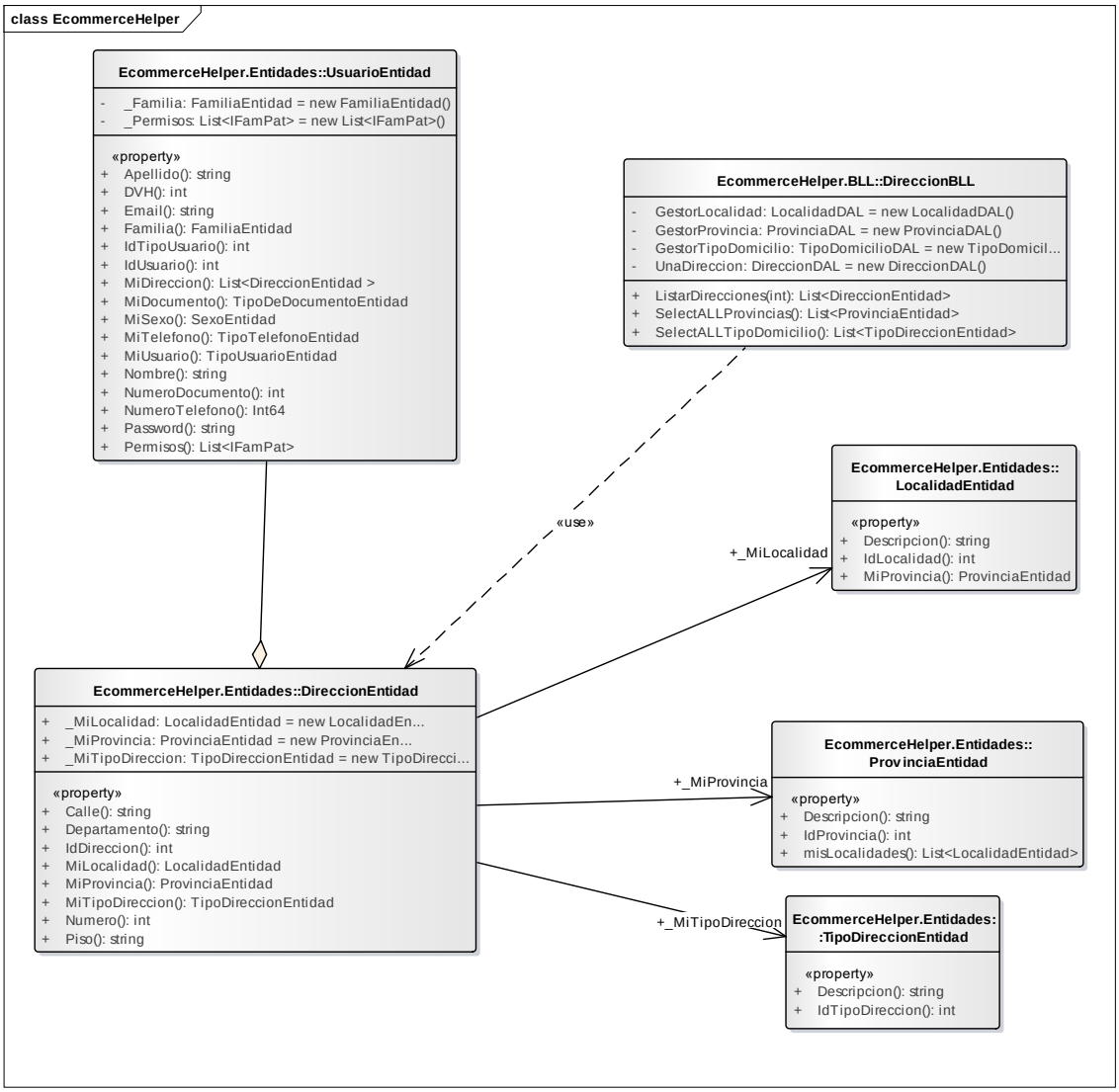
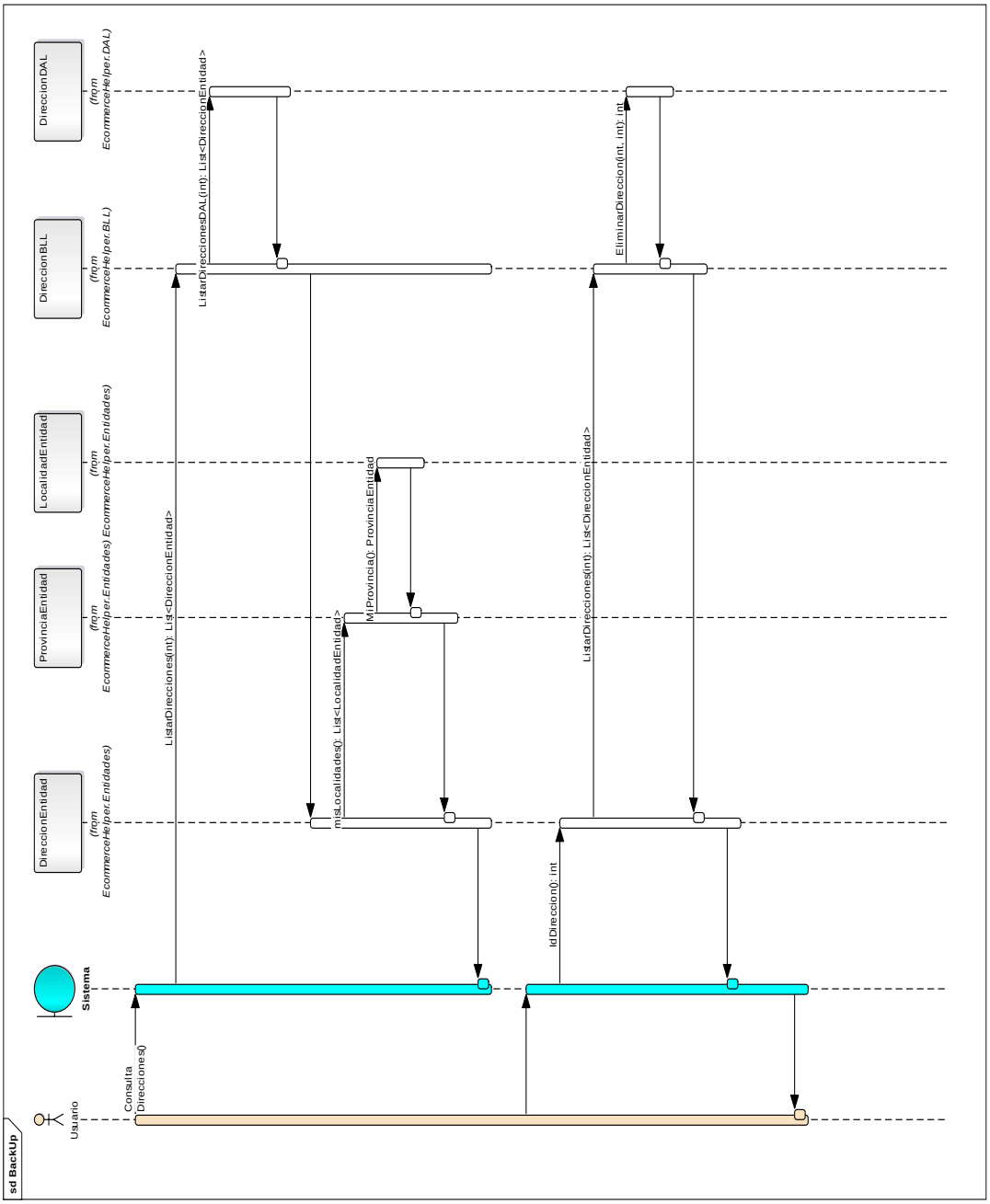

class EcommerceHelper
EcommerceHelper.Entidades::UsuarioEntidad
- _Familia: FamiliaEntidad = new FamiliaEntidad()
- _Permisos: List<IFamPat> = new List<IFamPat>()
«property»
+ Apellido(): string
+ DVH(): int
+ Email(): string
+ Familia(): FamiliaEntidad
+ IdTipoUsuario(): int
+ IdUsuario(): int
+ MiDireccion(): List<DireccionEntidad >
+ MiDocumento(): TipoDeDocumentoEntidad
+ MiSexo(): SexoEntidad
+ MiTelefono(): TipoTelefonoEntidad
+ MiUsuario(): TipoUsuarioEntidad
+ Nombre(): string
+ NumeroDocumento(): int
+ NumeroTelefono(): Int64
+ Password(): string
+ Permisos(): List<IFamPat>
EcommerceHelper.BLL::DireccionBLL
- GestorLocalidad: LocalidadDAL = new LocalidadDAL()
- GestorProvincia: ProvinciaDAL = new ProvinciaDAL()
- GestorTipoDomicilio: TipoDomicilioDAL = new TipoDomicil...
- UnaDireccion: DireccionDAL = new DireccionDAL()
+ ListarDirecciones(int): List<DireccionEntidad>
+ SelectALLProvincias(): List<ProvinciaEntidad>
+ SelectALLTipoDomicilio(): List<TipoDireccionEntidad>
«use»
EcommerceHelper.Entidades::LocalidadEntidad
«property»
+ Descripcion(): string
+ IdLocalidad(): int
+ MiProvincia(): ProvinciaEntidad
+_MiLocalidad
EcommerceHelper.Entidades::DireccionEntidad
+ _MiLocalidad: LocalidadEntidad = new LocalidadEn...
+ _MiProvincia: ProvinciaEntidad = new ProvinciaEn...
+ _MiTipoDireccion: TipoDireccionEntidad = new TipoDirecci...
«property»
+ Calle(): string
+ Departamento(): string
+ IdDireccion(): int
+ MiLocalidad(): LocalidadEntidad
+ MiProvincia(): ProvinciaEntidad
+ MiTipoDireccion(): TipoDireccionEntidad
+ Numero(): int
+ Piso(): string
+_MiProvincia
EcommerceHelper.Entidades::ProvinciaEntidad
«property»
+ Descripcion(): string
+ IdProvincia(): int
+ misLocalidades(): List<LocalidadEntidad>
+_MiTipoDireccion
EcommerceHelper.Entidades::TipoDireccionEntidad
«property»
+ Descripcion(): string
+ IdTipoDireccion(): int

Diagrama de Secuencia CU - 033 – Eliminar dirección



Pantalla

Calle	Numero	Piso	Departamento	Localidad	Provincia	Tipo Domicilio		
avellaneda	1078	0	0	La Reja	Buenos Aires	Quinta	Seleccionar	Eliminar
Int. Luis Tulissi	2348	1	a	Moreno	Buenos Aires	Departamento	Seleccionar	Eliminar

CU - 034 – Alta Tarjeta

Precondición
El usuario debe estar logueado con los permisos correspondientes para poder acceder al alta de una tarjeta, Usuario Cliente.

Actores primarios:
Usuario Cliente

Camino principal del caso de uso – Alta Tarjeta

1. El usuario cliente selecciona Registrar / Actualizar Tarjeta.
2. El sistema despliega el formulario.
3. El sistema solicita se ingrese:
    a. Titular de la tarjeta
    b. Numero de la Tarjeta
    c. Fecha de Vencimiento.
    d. Código de seguridad.

4. El usuario cliente ingresa los datos solicitados y presiona el botón Nuevo confirmando la operación alta Tarjeta.
5. El sistema valida los datos.
6. El sistema da de alta la tarjeta en el sistema.
7. El sistema refresca la lista de tarjeta y lo muestra en la grilla.

Caminos alternativos

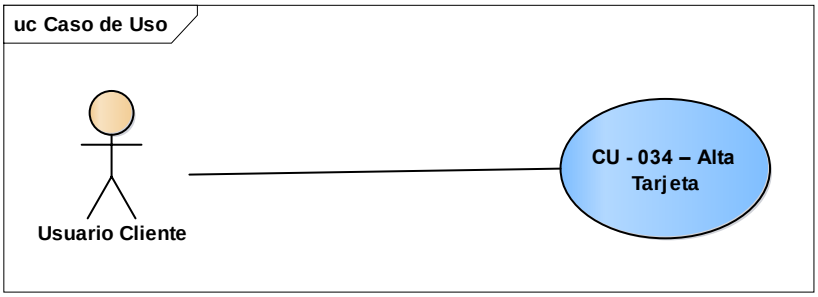
Para el paso 4: el sistema detecta que falta ingresar información o la misma no es del tipo indicado

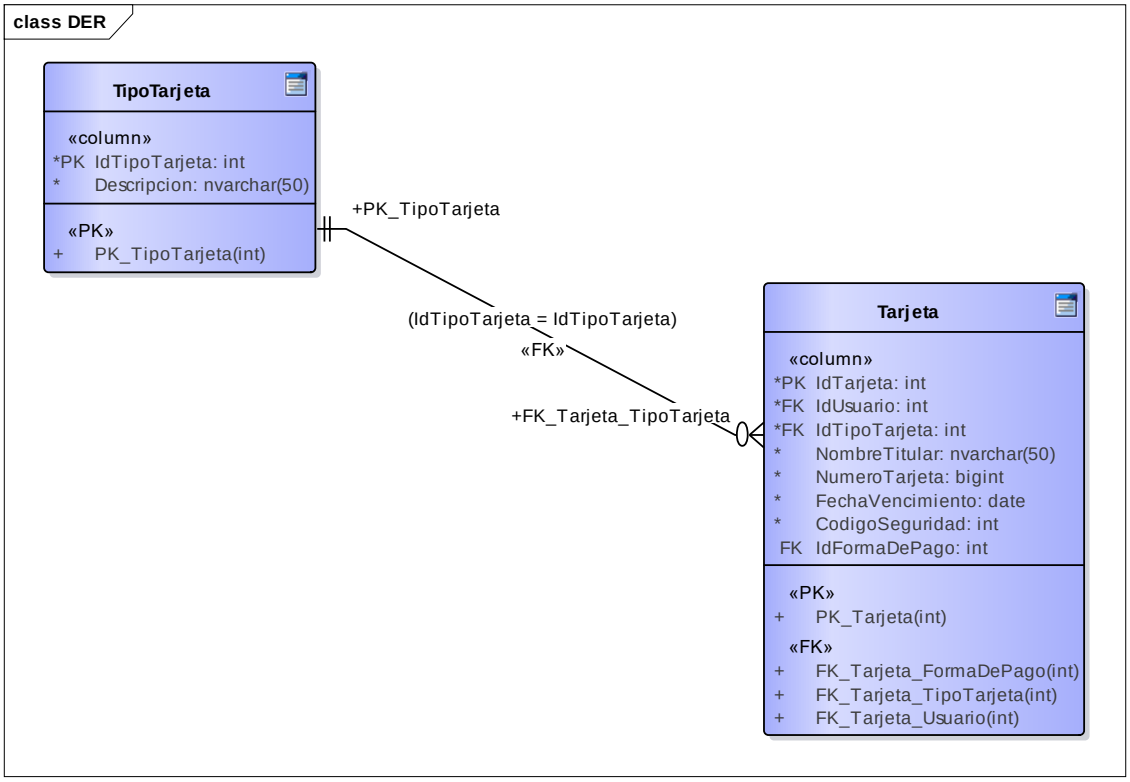
4.1 El sistema informa que aún falta ingresar datos o los datos ingresados no son del tipo indicado.

Flujo alternativo

Falla en el momento de grabación del alta de servicio.

Postcondición
La nueva tarjeta se encuentra creada.


uc Caso de Uso
Usuario Cliente
CU - 034 – Alta Tarjeta


class DER
TipoTarjeta
«column»
*PK IdTipoTarjeta: int
* Descripcion: nvarchar(50)
«PK»
+ PK_TipoTarjeta(int)
+PK_TipoTarjeta
(IdTipoTarjeta = IdTipoTarjeta)
«FK»
+FK_Tarjeta_TipoTarjeta
Tarjeta
«column»
*PK IdTarjeta: int
*FK IdUsuario: int
*FK IdTipoTarjeta: int
* NombreTitular: nvarchar(50)
* NumeroTarjeta: bigint
* FechaVencimiento: date
* CodigoSeguridad: int
FK IdFormaDePago: int
«PK»
+ PK_Tarjeta(int)
«FK»
+ FK_Tarjeta_FormaDePago(int)
+ FK_Tarjeta_TipoTarjeta(int)
+ FK_Tarjeta_Usuario(int)

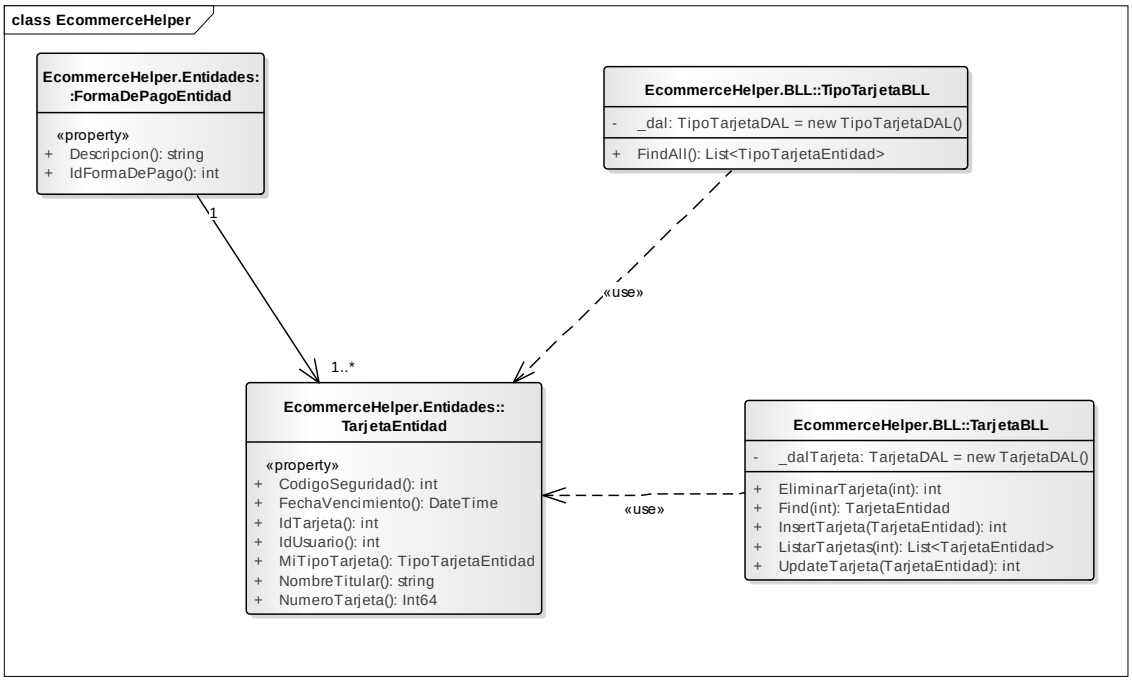
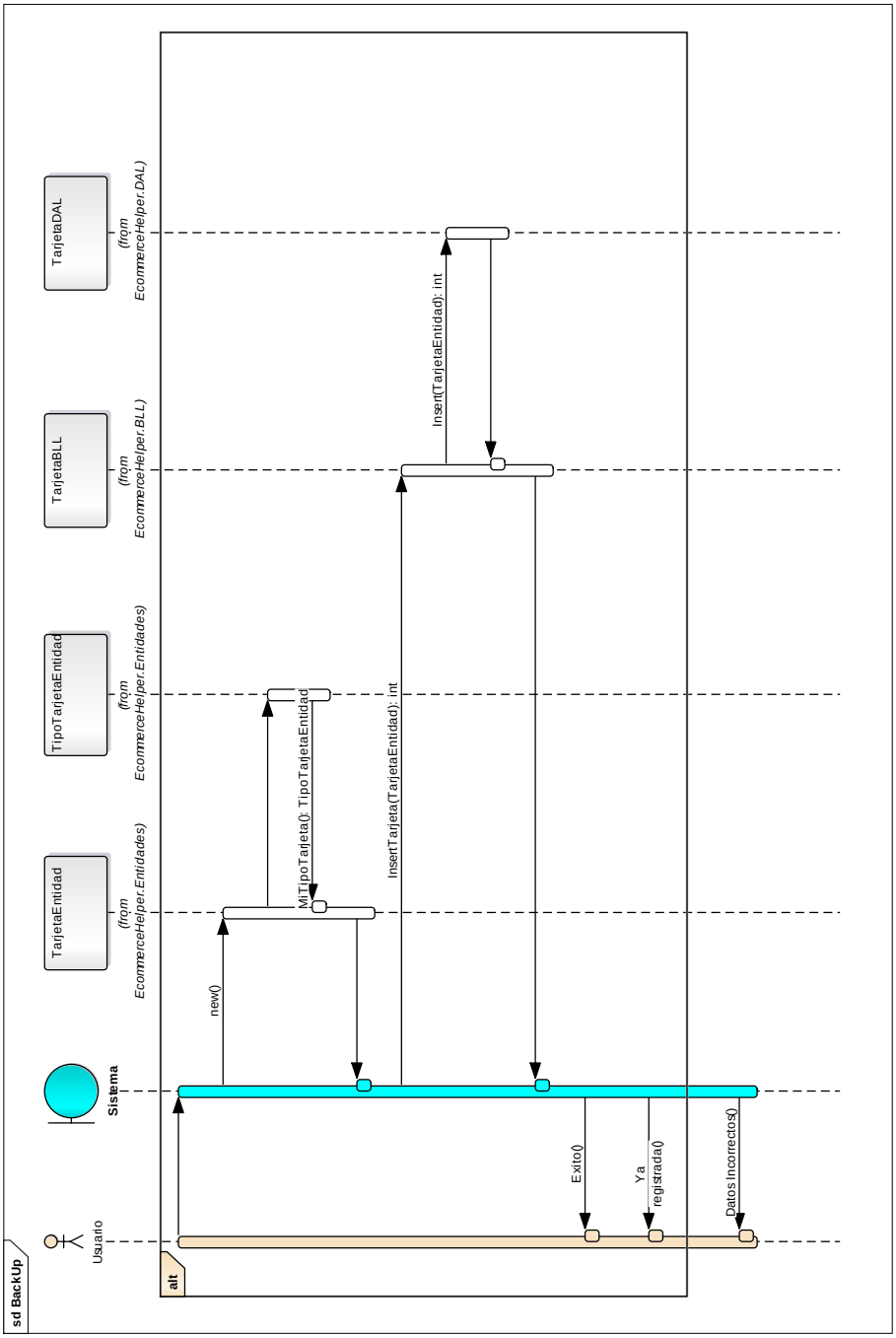

class EcommerceHelper
EcommerceHelper.Entidades::FormaDePagoEntidad
«property»
+ Descripcion(): string
+ IdFormaDePago(): int
EcommerceHelper.BLL::TipoTarjetaBLL
- _dal: TipoTarjetaDAL = new TipoTarjetaDAL()
+ FindAll(): List<TipoTarjetaEntidad>
1
1..*
«use»
EcommerceHelper.Entidades::TarjetaEntidad
«property»
+ CodigoSeguridad(): int
+ FechaVencimiento(): DateTime
+ IdTarjeta(): int
+ IdUsuario(): int
+ MiTipoTarjeta(): TipoTarjetaEntidad
+ NombreTitular(): string
+ NumeroTarjeta(): Int64
«use»
EcommerceHelper.BLL::TarjetaBLL
- _dalTarjeta: TarjetaDAL = new TarjetaDAL()
+ EliminarTarjeta(int): int
+ Find(int): TarjetaEntidad
+ InsertTarjeta(TarjetaEntidad): int
+ ListarTarjetas(int): List<TarjetaEntidad>
+ UpdateTarjeta(TarjetaEntidad): int

Diagrama de Secuencia CU - 034 – Alta Tarjeta



Pantalla

Titular

Esteban Ensueño Pesadilla

Numero

4398189693448976

Fecha De Vencimiento

dd/mm/aaaa

Codigo de Seguridad

111

Tipo de Tarjeta

Debito

Alta | Guardar Cambios | Volver A Pago

CU - 035 – Modificar Tarjeta

Precondición
El usuario debe estar logueado con los permisos correspondientes para poder acceder a modificar una tarjeta, Usuario Cliente.

Actores primarios:
Usuario Cliente

Camino principal del caso de uso – Modificar Tarjeta

1. El usuario cliente selecciona Registrar / Actualizar Tarjeta.
2. El sistema despliega el formulario.
3. El usuario selecciona la tarjeta a modificar.
4. El sistema despliega en el formulario los datos de la tarjeta a actualizar:
    a. Titular de la tarjeta
    b. Numero de la Tarjeta
    c. Fecha de Vencimiento.
    d. Código de seguridad.

5. El usuario cliente ingresa los datos a modificar y presiona el botón Guardar Cambios confirmando la operación modificación de la Tarjeta.
6. El sistema valida los datos.
7. El sistema modifica la tarjeta en el sistema.
8. El sistema refresca la lista de tarjeta y lo muestra en la grilla.

Caminos alternativos

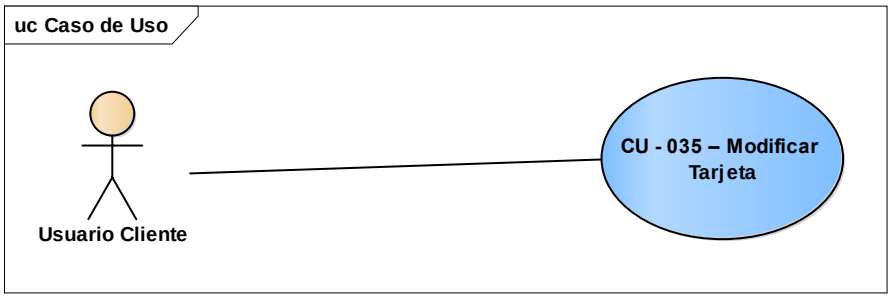
Para el paso 4: el sistema detecta que falta ingresar información o la misma no es del tipo indicado

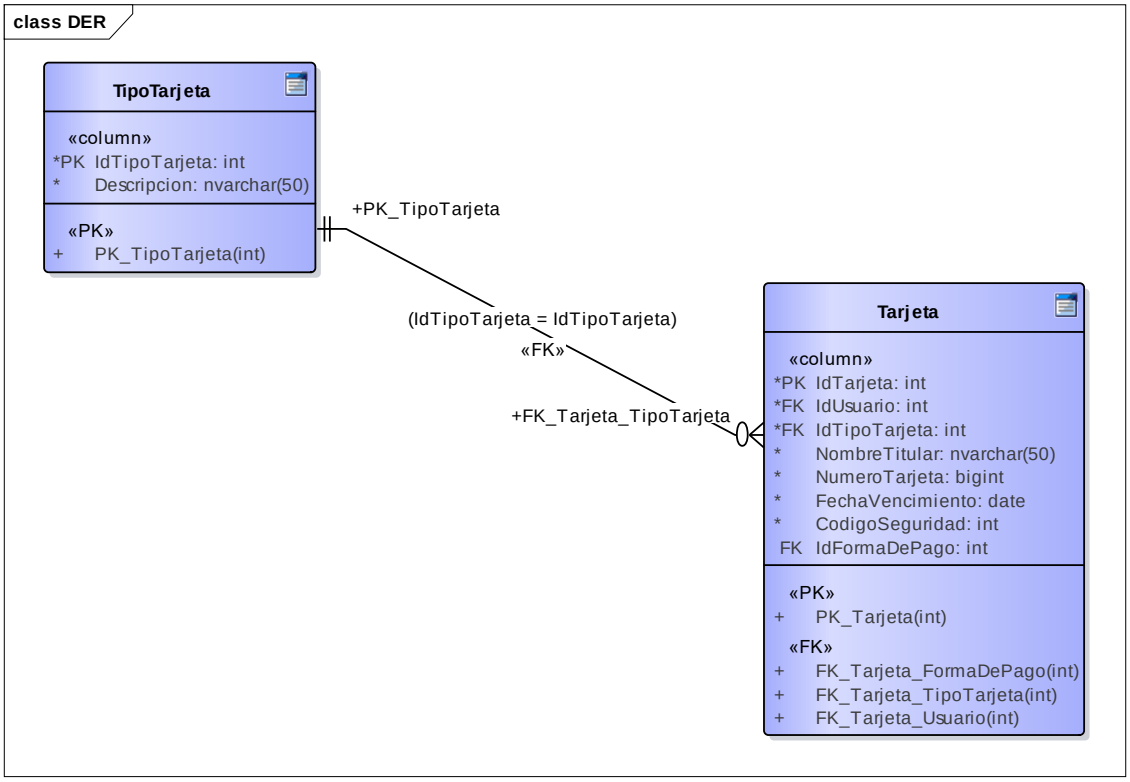
4.1 El sistema informa que aún falta ingresar datos o los datos ingresados no son del tipo indicado.

Flujo alternativo

Falla en el momento de grabación del alta de servicio.

Postcondición
La tarjeta se encuentra modificada.


uc Caso de Uso
Usuario Cliente
CU - 035 – Modificar Tarjeta


class DER
TipoTarjeta
«column»
*PK IdTipoTarjeta: int
* Descripcion: nvarchar(50)
«PK»
+ PK_TipoTarjeta(int)
+PK_TipoTarjeta
(IdTipoTarjeta = IdTipoTarjeta)
«FK»
+FK_Tarjeta_TipoTarjeta
Tarjeta
«column»
*PK IdTarjeta: int
*FK IdUsuario: int
*FK IdTipoTarjeta: int
* NombreTitular: nvarchar(50)
* NumeroTarjeta: bigint
* FechaVencimiento: date
* CodigoSeguridad: int
FK IdFormaDePago: int
«PK»
+ PK_Tarjeta(int)
«FK»
+ FK_Tarjeta_FormaDePago(int)
+ FK_Tarjeta_TipoTarjeta(int)
+ FK_Tarjeta_Usuario(int)

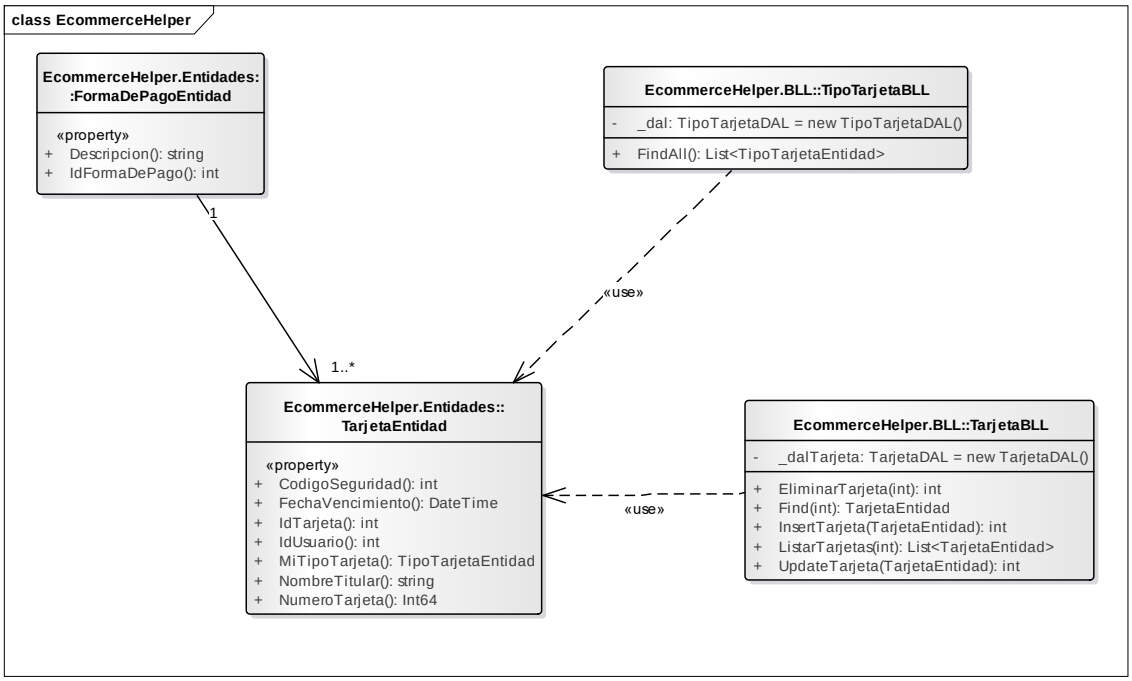
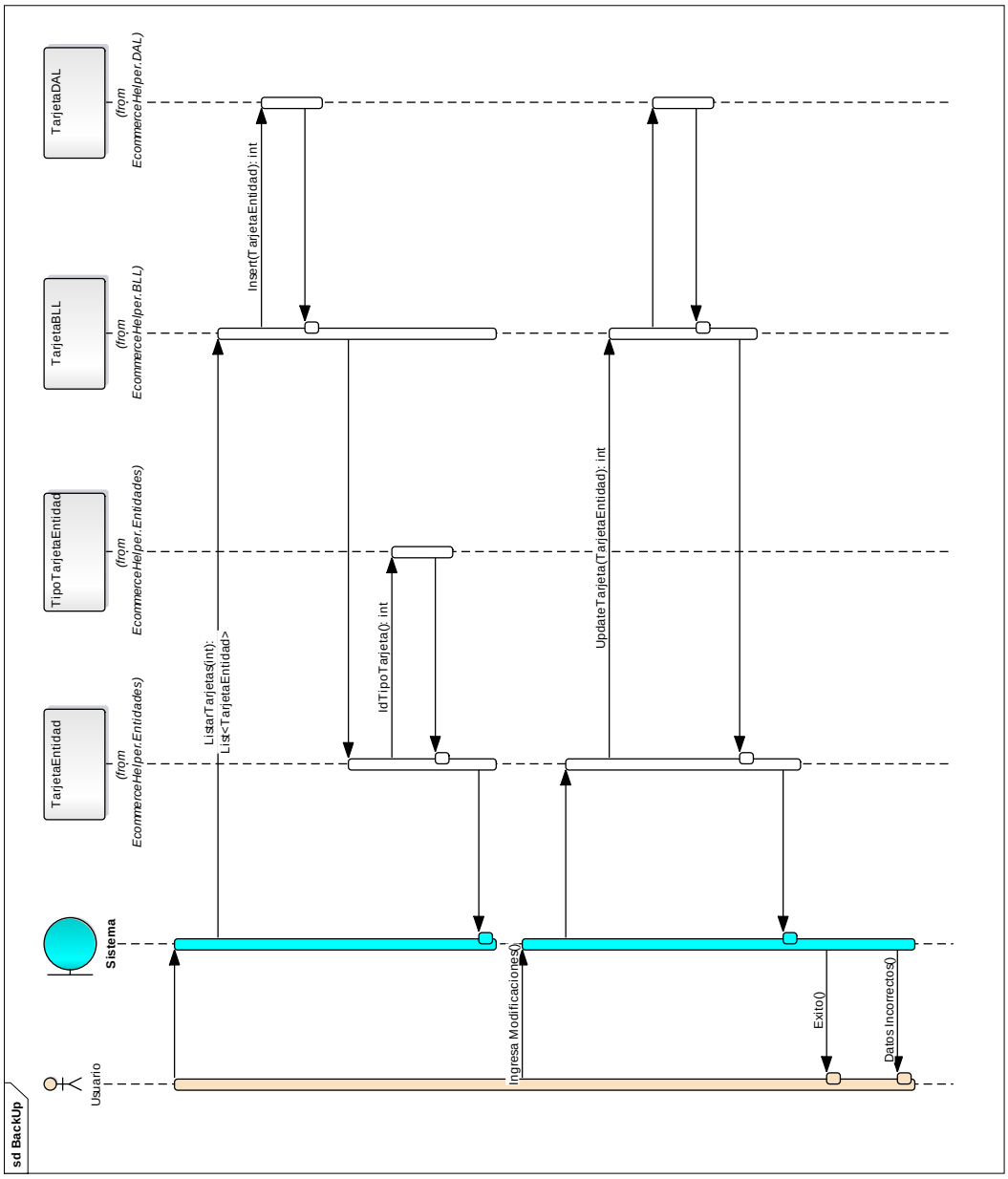

class EcommerceHelper
EcommerceHelper.Entidades::FormaDePagoEntidad
«property»
+ Descripcion(): string
+ IdFormaDePago(): int
EcommerceHelper.BLL::TipoTarjetaBLL
- _dal: TipoTarjetaDAL = new TipoTarjetaDAL()
+ FindAll(): List<TipoTarjetaEntidad>
1
1..*
«use»
EcommerceHelper.Entidades::TarjetaEntidad
«property»
+ CodigoSeguridad(): int
+ FechaVencimiento(): DateTime
+ IdTarjeta(): int
+ IdUsuario(): int
+ MiTipoTarjeta(): TipoTarjetaEntidad
+ NombreTitular(): string
+ NumeroTarjeta(): Int64
«use»
EcommerceHelper.BLL::TarjetaBLL
- _dalTarjeta: TarjetaDAL = new TarjetaDAL()
+ EliminarTarjeta(int): int
+ Find(int): TarjetaEntidad
+ InsertTarjeta(TarjetaEntidad): int
+ ListarTarjetas(int): List<TarjetaEntidad>
+ UpdateTarjeta(TarjetaEntidad): int

Diagrama de Secuencia CU - 035 – Modificar Tarjeta



Pantalla

Titular

Esteban Ensueño Pesadilla

Numero

4398189693448976

Fecha De Vencimiento

dd/mm/aaaa

Codigo de Seguridad

111

Tipo de Tarjeta

Debito

Alta | Guardar Cambios | Volver A Pago

Titular	Numero	Tipo Tarjeta	Fecha de Vencimiento	Codigo de Seguridad		
Esteban Ensueño Pesadilla	4398189693448976	Credito	30/11/2020 00:00:00	111	Modificar	Eliminar

CU - 036 – Baja Tarjeta

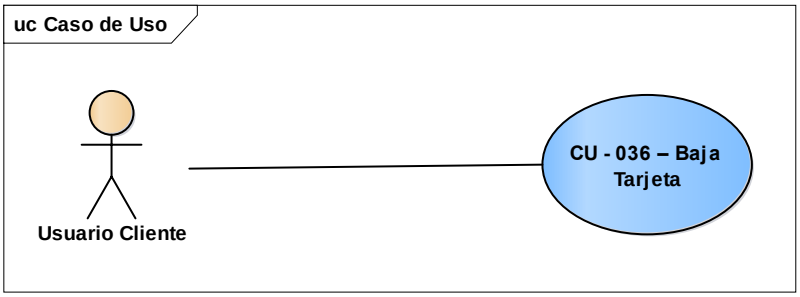
Precondición
El usuario debe estar logueado con los permisos correspondientes para poder acceder al dar de baja una tarjeta, Usuario Cliente.

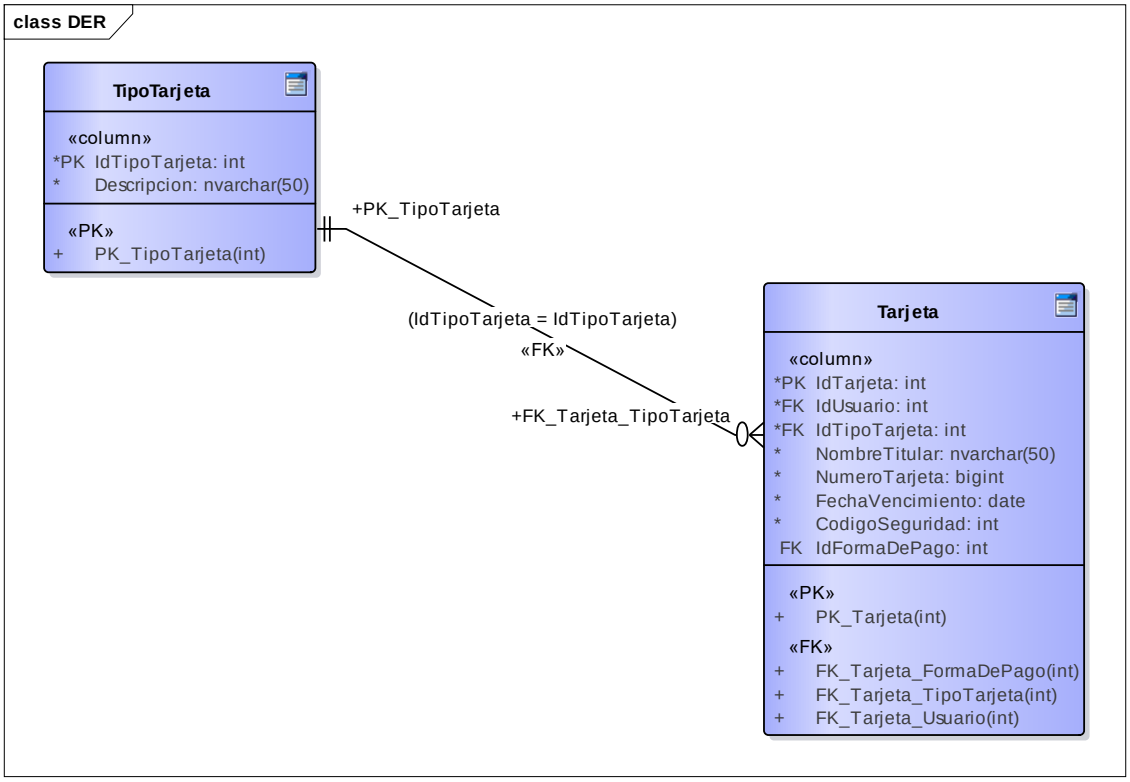
Actores primarios:
Usuario Cliente

Camino principal del caso de uso – Eliminar Tarjeta

1. El usuario cliente selecciona de la grilla de tarjetas el que desea eliminar.
2. El sistema elimina la tarjeta.
3. El sistema refresca la grilla y actualiza las tarjetas disponibles.

Postcondición
La tarjeta se encuentra eliminada.


uc Caso de Uso
Usuario Cliente
CU - 036 – Baja Tarjeta


class DER
TipoTarjeta
«column»
*PK IdTipoTarjeta: int
* Descripcion: nvarchar(50)
«PK»
+ PK_TipoTarjeta(int)
+PK_TipoTarjeta
(IdTipoTarjeta = IdTipoTarjeta)
«FK»
+FK_Tarjeta_TipoTarjeta
Tarjeta
«column»
*PK IdTarjeta: int
*FK IdUsuario: int
*FK IdTipoTarjeta: int
* NombreTitular: nvarchar(50)
* NumeroTarjeta: bigint
* FechaVencimiento: date
* CodigoSeguridad: int
FK IdFormaDePago: int
«PK»
+ PK_Tarjeta(int)
«FK»
+ FK_Tarjeta_FormaDePago(int)
+ FK_Tarjeta_TipoTarjeta(int)
+ FK_Tarjeta_Usuario(int)

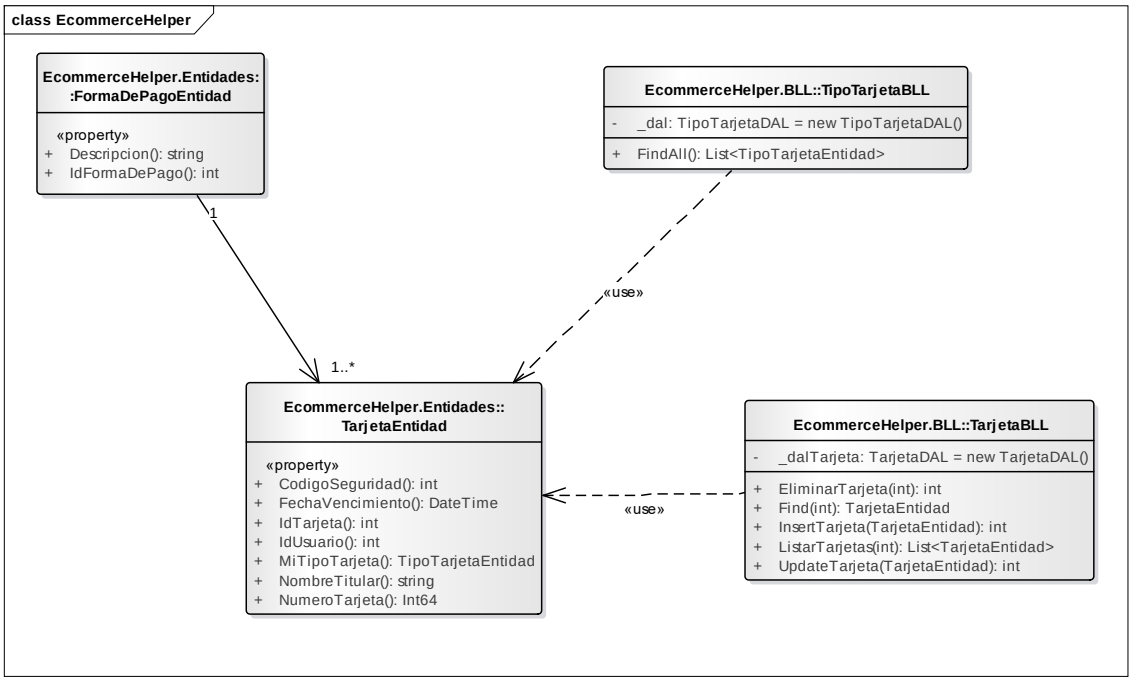
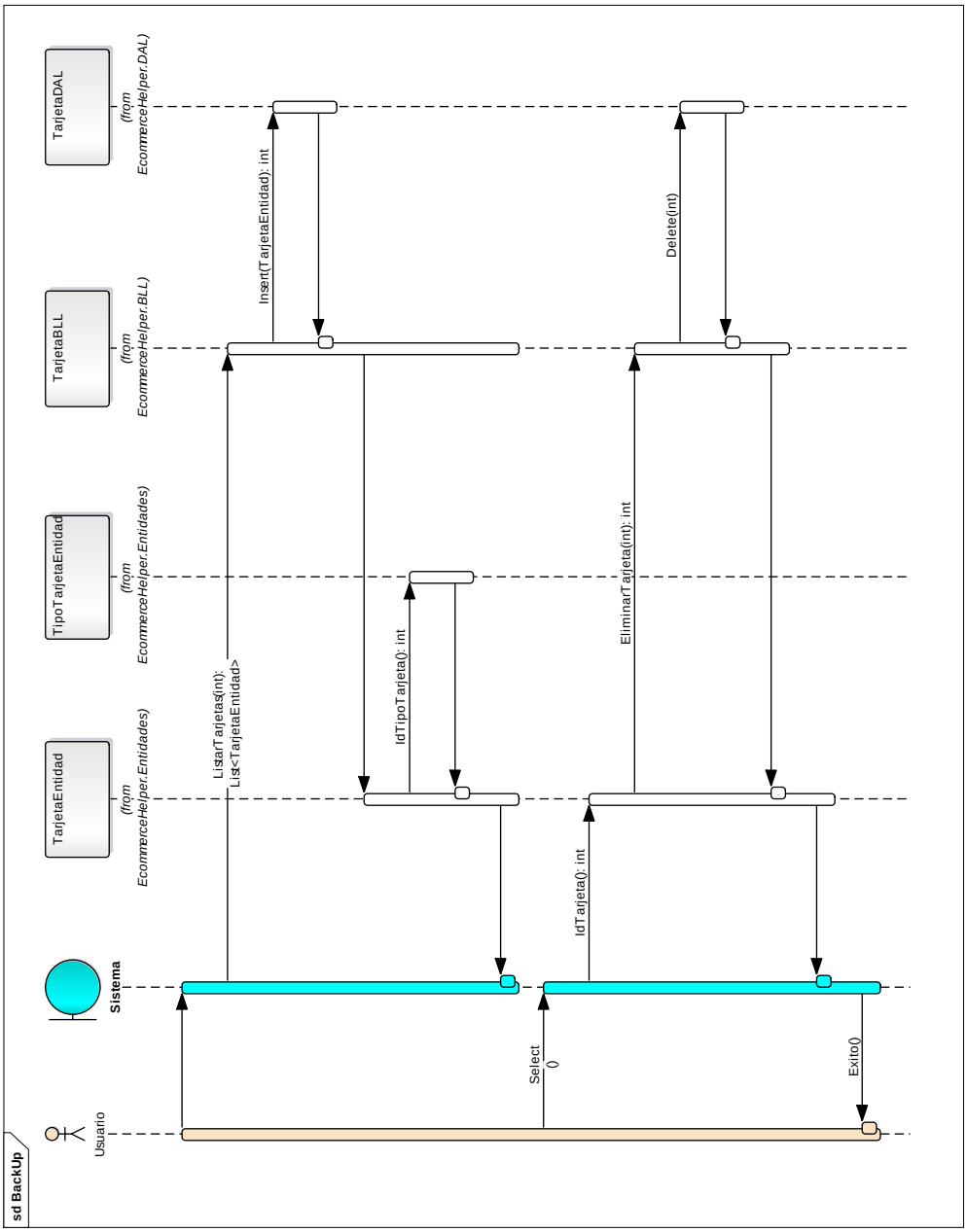

class EcommerceHelper
EcommerceHelper.Entidades::FormaDePagoEntidad
«property»
+ Descripcion(): string
+ IdFormaDePago(): int
EcommerceHelper.BLL::TipoTarjetaBLL
- _dal: TipoTarjetaDAL = new TipoTarjetaDAL()
+ FindAll(): List<TipoTarjetaEntidad>
1
1..*
«use»
EcommerceHelper.Entidades::TarjetaEntidad
«property»
+ CodigoSeguridad(): int
+ FechaVencimiento(): DateTime
+ IdTarjeta(): int
+ IdUsuario(): int
+ MiTipoTarjeta(): TipoTarjetaEntidad
+ NombreTitular(): string
+ NumeroTarjeta(): Int64
«use»
EcommerceHelper.BLL::TarjetaBLL
- _dalTarjeta: TarjetaDAL = new TarjetaDAL()
+ EliminarTarjeta(int): int
+ Find(int): TarjetaEntidad
+ InsertTarjeta(TarjetaEntidad): int
+ ListarTarjetas(int): List<TarjetaEntidad>
+ UpdateTarjeta(TarjetaEntidad): int

Diagrama de Secuencia CU - 036 – Baja Tarjeta



Pantalla

Titular	Numero	Tipo Tarjeta	Fecha de Vencimiento	Codigo de Seguridad		
Esteban Ensueño Pesadilla	4398189693448976	Credito	30/11/2020 00:00:00	111	Modificar	Eliminar

CU - 037 – Recuperar Contraseña

Precondición
No existe precondición.

Actores primarios:
Usuario

Camino principal del caso de uso – Recuperar contraseña

1. El usuario selecciona Recuperar Contraseña
2. El sistema despliega el formulario.
3. El usuario ingresa su correo electrónico.
4. El sistema verifica la existencia del usuario.
5. El sistema genera automáticamente una contraseña alfabética diferenciando mayúsculas de minúsculas.
6. El sistema encripta la nueva contraseña.
7. El sistema actualiza la contraseña del usuario en los registros de la base de datos.
8. El sistema envía un mail con la nueva contraseña al usuario.
9. El sistema retorna al usuario a la página de iniciar sesión.

Caminos alternativos

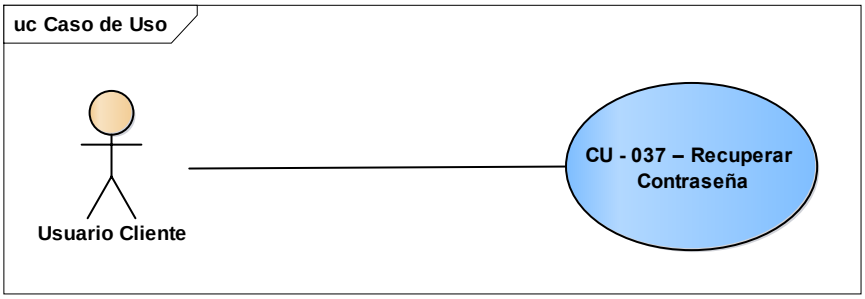
Para el paso 4: el sistema detecta que el usuario no existe.

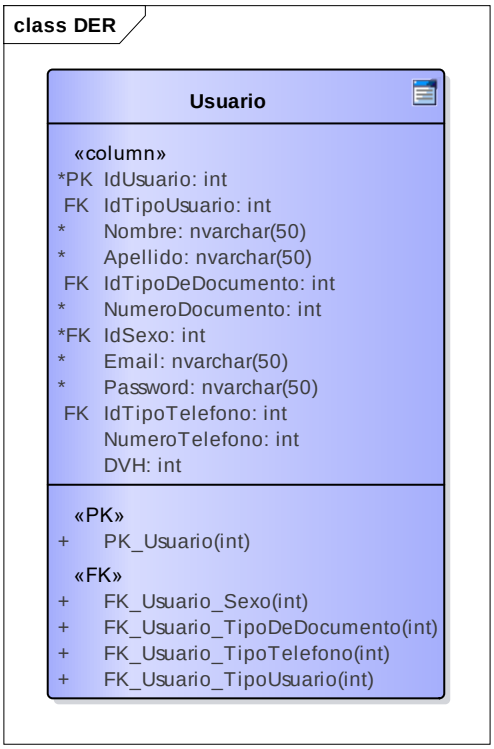
4.1 El sistema informa el correo electrónico ingresado no corresponde a un usuario.

Flujo alternativo

El usuario presiona el botón Cancelar.
El sistema retorna a la pantalla de inicio.

Postcondición
La contraseña se encuentra modificada.


uc Caso de Uso
Usuario Cliente
CU - 037 – Recuperar Contraseña


class DER
Usuario
«column»
*PK IdUsuario: int
FK IdTipoUsuario: int
* Nombre: nvarchar(50)
* Apellido: nvarchar(50)
FK IdTipoDeDocumento: int
* NumeroDocumento: int
*FK IdSexo: int
* Email: nvarchar(50)
* Password: nvarchar(50)
FK IdTipoTelefono: int
NumeroTelefono: int
DVH: int
«PK»
+ PK_Usuario(int)
«FK»
+ FK_Usuario_Sexo(int)
+ FK_Usuario_TipoDeDocumento(int)
+ FK_Usuario_TipoTelefono(int)
+ FK_Usuario_TipoUsuario(int)

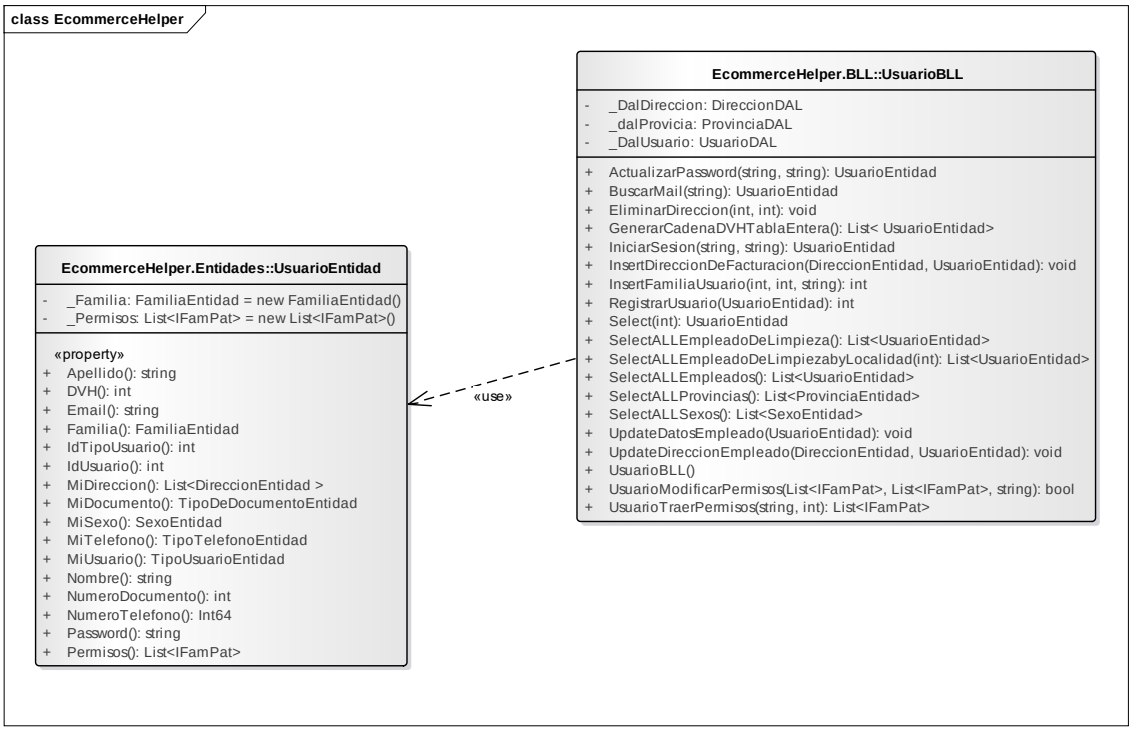
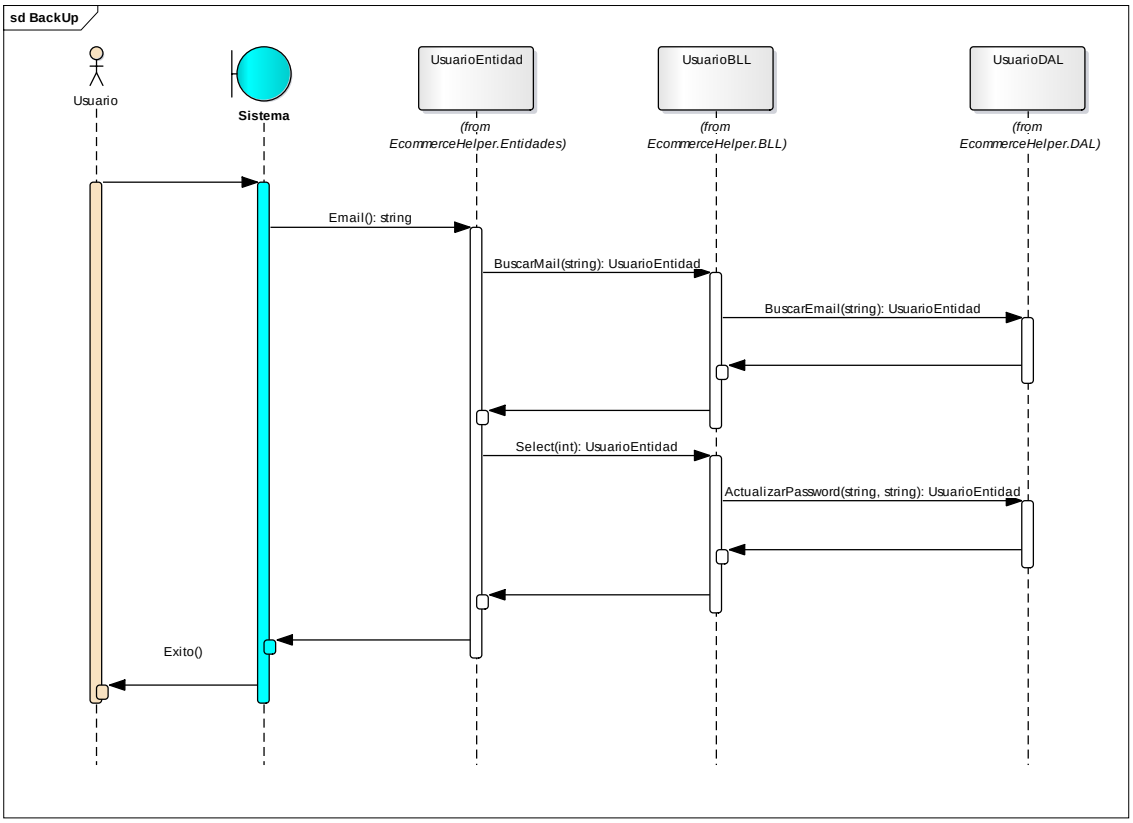

class EcommerceHelper
EcommerceHelper.BLL::UsuarioBLL
- _DalDireccion: DireccionDAL
- _dalProvicia: ProvinciaDAL
- _DalUsuario: UsuarioDAL
+ ActualizarPassword(string, string): UsuarioEntidad
+ BuscarMail(string): UsuarioEntidad
+ EliminarDireccion(int, int): void
+ GenerarCadenaDVHTablaEntera(): List< UsuarioEntidad>
+ IniciarSesion(string, string): UsuarioEntidad
+ InsertDireccionDeFacturacion(DireccionEntidad, UsuarioEntidad): void
+ InsertFamiliaUsuario(int, int, string): int
+ RegistrarUsuario(UsuarioEntidad): int
+ Select(int): UsuarioEntidad
+ SelectALLEmpleadoDeLimpieza(): List<UsuarioEntidad>
+ SelectALLEmpleadoDeLimpiezabyLocalidad(int): List<UsuarioEntidad>
+ SelectALLEmpleados(): List<UsuarioEntidad>
+ SelectALLProvincias(): List<ProvinciaEntidad>
+ SelectALLSexos(): List<SexoEntidad>
+ UpdateDatosEmpleado(UsuarioEntidad): void
+ UpdateDireccionEmpleado(DireccionEntidad, UsuarioEntidad): void
+ UsuarioBLL()
+ UsuarioModificarPermisos(List<IFamPat>, List<IFamPat>, string): bool
+ UsuarioTraerPermisos(string, int): List<IFamPat>
«use»
EcommerceHelper.Entidades::UsuarioEntidad
- _Familia: FamiliaEntidad = new FamiliaEntidad()
- _Permisos: List<IFamPat> = new List<IFamPat>()
«property»
+ Apellido(): string
+ DVH(): int
+ Email(): string
+ Familia(): FamiliaEntidad
+ IdTipoUsuario(): int
+ IdUsuario(): int
+ MiDireccion(): List<DireccionEntidad >
+ MiDocumento(): TipoDeDocumentoEntidad
+ MiSexo(): SexoEntidad
+ MiTelefono(): TipoTelefonoEntidad
+ MiUsuario(): TipoUsuarioEntidad
+ Nombre(): string
+ NumeroDocumento(): int
+ NumeroTelefono(): Int64
+ Password(): string
+ Permisos(): List<IFamPat>

Diagrama de Secuencia CU - 037 – Recuperar Contraseña



Pantalla

RecuperarContraseña
Usuario
Ingrese mail ejemplo@dominio.com
Aceptar
Cancelar

Requisitos relacionados a Personal

- CU - 007 – Alta Empleado
- CU - 008 – Asignación de Perfiles
- CU – 030 – Modificar Empleado
- CU – 031 – Eliminar Empleado
- CU - 009 – Consultar Empleado

CU - 007 – Alta Empleado

Precondición
El personal de recursos humanos debe estar logueado.

Actores primarios:
Recursos humanos.

Camino principal del caso de uso – Alta Empleado

1. El usuario de recursos humanos selecciona dar de alta un nuevo empleado.
2. El sistema solicita se ingrese:
    a. E-mail: Texto con formato ejemplo@ejemplo.com
    b. Contraseña (alfanumérica ce 8 caracteres que incluya mayúsculas y minúsculas)
    c. Reingrese contraseña.

3. El sistema solicita los demás datos básicos:
    a. Nombres o Nombre: Texto
    b. Apellidos: Texto.
    c. Sexo
    d. DNI
    e. Tipo de teléfono
    f. Número de teléfono (mínimo 8 dígitos) 1234-5678
    g. Dirección

        i. Provincia: texto
        ii. Localidad: texto
        iii. Calle: texto
        iv. Nro. de calle: Números
        v. Piso: Número.
        vi. Departamento: texto
        vii. Tipo de Domicilio

4. El usuario cliente ingresa los datos solicitados y confirma la operación crear cuenta.
5. El sistema valida los datos.
6. El sistema verifica que el usuario ya no esté registrado.
7. El sistema encripta la contraseña.

8. El sistema genera el Digito Verificador horizontal.
9. El sistema registra el nuevo usuario empleado.
10. El sistema registra en la bitácora el evento de alta de nuevo empleado.
11. El sistema actualiza el valor del Digito Verificador Vertical tabla Usuario.

Caminos alternativos

Para el paso 5: el sistema detecta que falta ingresar información o la misma no es del tipo indicado

5.1 El sistema informa que aún falta ingresar datos o los datos ingresados no son del tipo indicado.

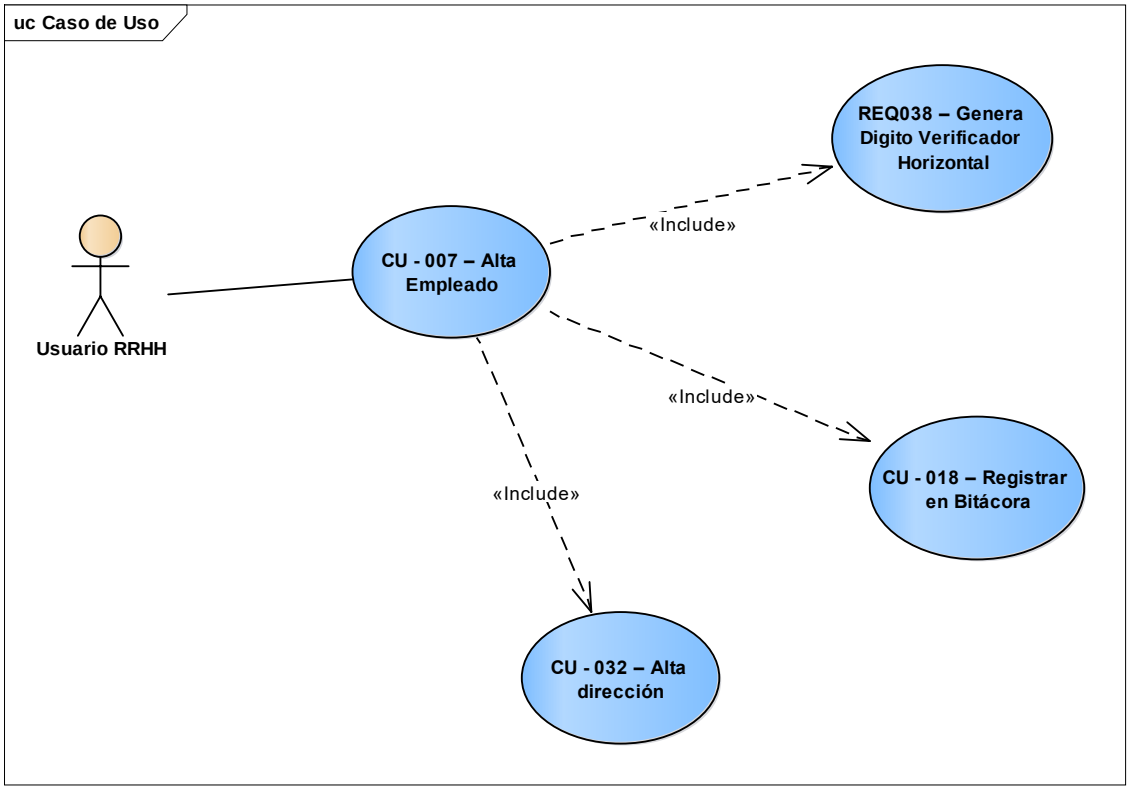
Para el paso 6: La dirección de correo electrónico ya se encuentra registrada

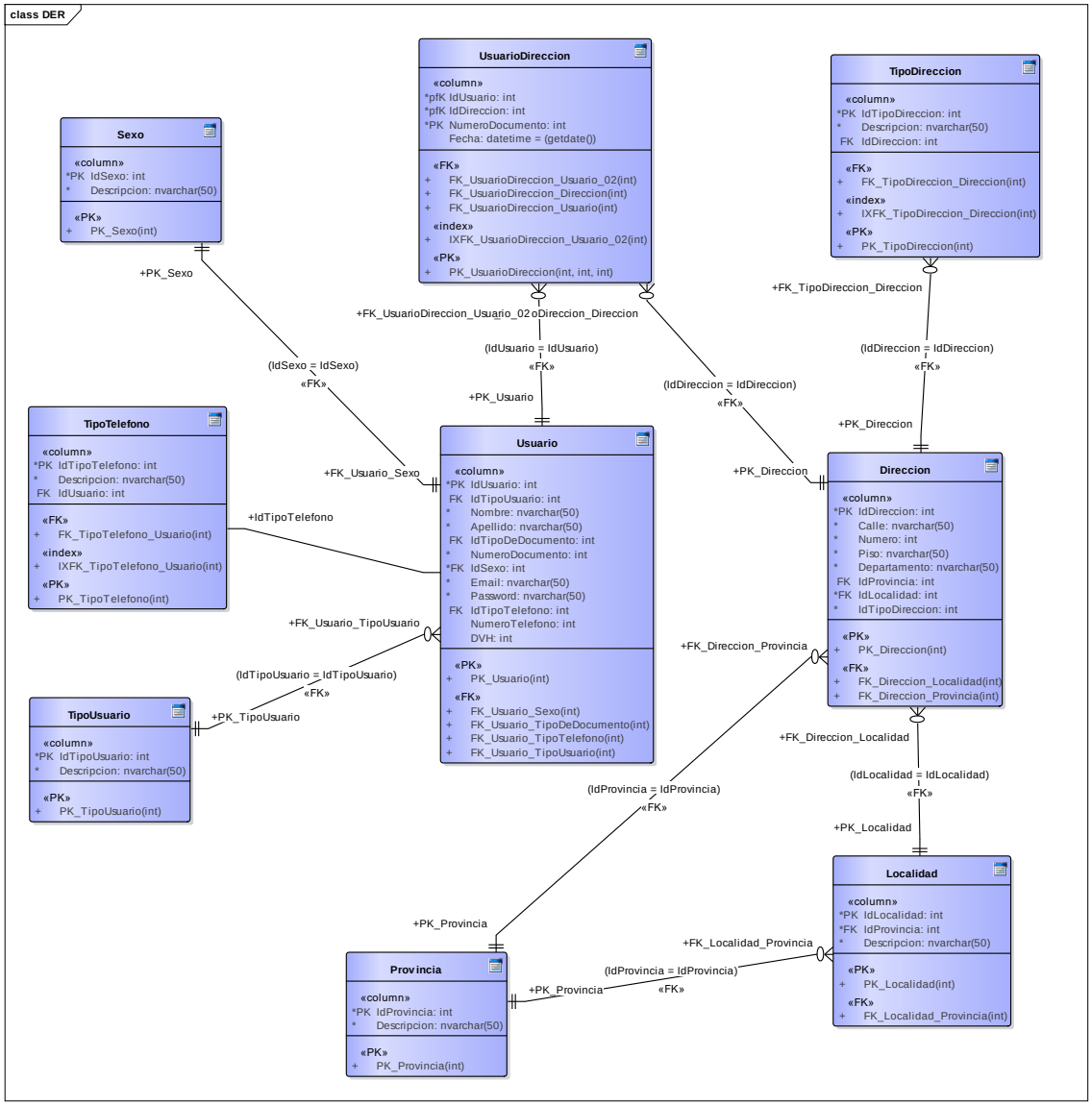
6.1 El sistema informa que la dirección de correo electrónica ya se encuentra registrada.

Postcondición
El cliente se encuentra registrado en la página.

Flujo alternativo

Falla en el momento de grabación del alta del nuevo empleado.
El sistema registra en bitácora el error producido.


uc Caso de Uso
REQ038 – Genera Digito Verificador Horizontal
«Include»
Usuario RRHH
CU - 007 – Alta Empleado
«Include»
CU - 018 – Registrar en Bitácora
«Include»
CU - 032 – Alta dirección


class DER
Sexo
«column»
*PK IdSexo: int
* Descripcion: nvarchar(50)
«PK»
+ PK_Sexo(int)
UsuarioDireccion
«column»
*pfK IdUsuario: int
*pfK IdDireccion: int
*PK NumeroDocumento: int
Fecha: datetime = (getdate())
«FK»
+ FK_UsuarioDireccion_Usuario_02(int)
+ FK_UsuarioDireccion_Direccion(int)
+ FK_UsuarioDireccion_Usuario(int)
«index»
+ IXFK_UsuarioDireccion_Usuario_02(int)
«PK»
+ PK_UsuarioDireccion(int, int, int)
TipoDireccion
«column»
*PK IdTipoDireccion: int
* Descripcion: nvarchar(50)
FK IdDireccion: int
«FK»
+ FK_TipoDireccion_Direccion(int)
«index»
+ IXFK_TipoDireccion_Direccion(int)
«PK»
+ PK_TipoDireccion(int)
+PK_Sexo
+FK_UsuarioDireccion_Usuario_02
+FK_TipoDireccion_Direccion
(IdSexo = IdSexo)
«FK»
(IdUsuario = IdUsuario)
«FK»
(IdDireccion = IdDireccion)
«FK»
(IdDireccion = IdDireccion)
«FK»
+PK_Usuario
+PK_Direccion
TipoTelefono
«column»
*PK IdTipoTelefono: int
* Descripcion: nvarchar(50)
FK IdUsuario: int
«FK»
+ FK_TipoTelefono_Usuario(int)
«index»
+ IXFK_TipoTelefono_Usuario(int)
«PK»
+ PK_TipoTelefono(int)
+FK_Usuario_Sexo
+PK_Direccion
+IdTipoTelefono
Usuario
«column»
*PK IdUsuario: int
FK IdTipoUsuario: int
* Nombre: nvarchar(50)
* Apellido: nvarchar(50)
FK IdTipoDeDocumento: int
* NumeroDocumento: int
*FK IdSexo: int
* Email: nvarchar(50)
* Password: nvarchar(50)
FK IdTipoTelefono: int
NumeroTelefono: int
DVH: int
«PK»
+ PK_Usuario(int)
«FK»
+ FK_Usuario_Sexo(int)
+ FK_Usuario_TipoDeDocumento(int)
+ FK_Usuario_TipoTelefono(int)
+ FK_Usuario_TipoUsuario(int)
Direccion
«column»
*PK IdDireccion: int
* Calle: nvarchar(50)
* Numero: int
* Piso: nvarchar(50)
* Departamento: nvarchar(50)
FK IdProvincia: int
*FK IdLocalidad: int
* IdTipoDireccion: int
«PK»
+ PK_Direccion(int)
«FK»
+ FK_Direccion_Localidad(int)
+ FK_Direccion_Provincia(int)
+FK_Usuario_TipoUsuario
(IdTipoUsuario = IdTipoUsuario)
«FK»
+FK_Direccion_Provincia
TipoUsuario
«column»
*PK IdTipoUsuario: int
* Descripcion: nvarchar(50)
«PK»
+ PK_TipoUsuario(int)
+PK_TipoUsuario
+FK_Direccion_Localidad
(IdProvincia = IdProvincia)
«FK»
(IdLocalidad = IdLocalidad)
«FK»
+PK_Localidad
Localidad
«column»
*PK IdLocalidad: int
*FK IdProvincia: int
* Descripcion: nvarchar(50)
«PK»
+ PK_Localidad(int)
«FK»
+ FK_Localidad_Provincia(int)
+PK_Provincia
+FK_Localidad_Provincia
(IdProvincia = IdProvincia)
«FK»
+PK_Provincia
Provincia
«column»
*PK IdProvincia: int
* Descripcion: nvarchar(50)
«PK»
+ PK_Provincia(int)

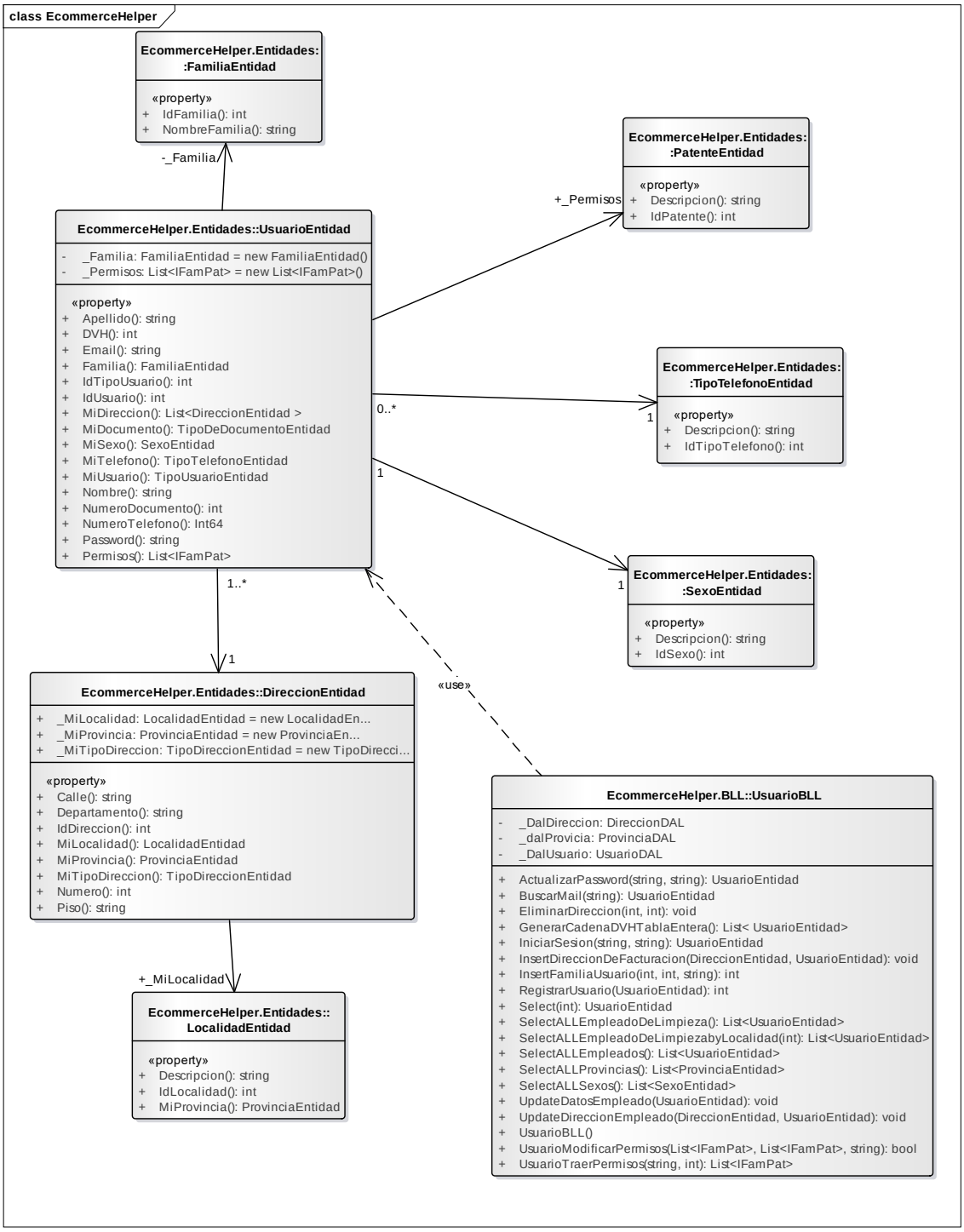
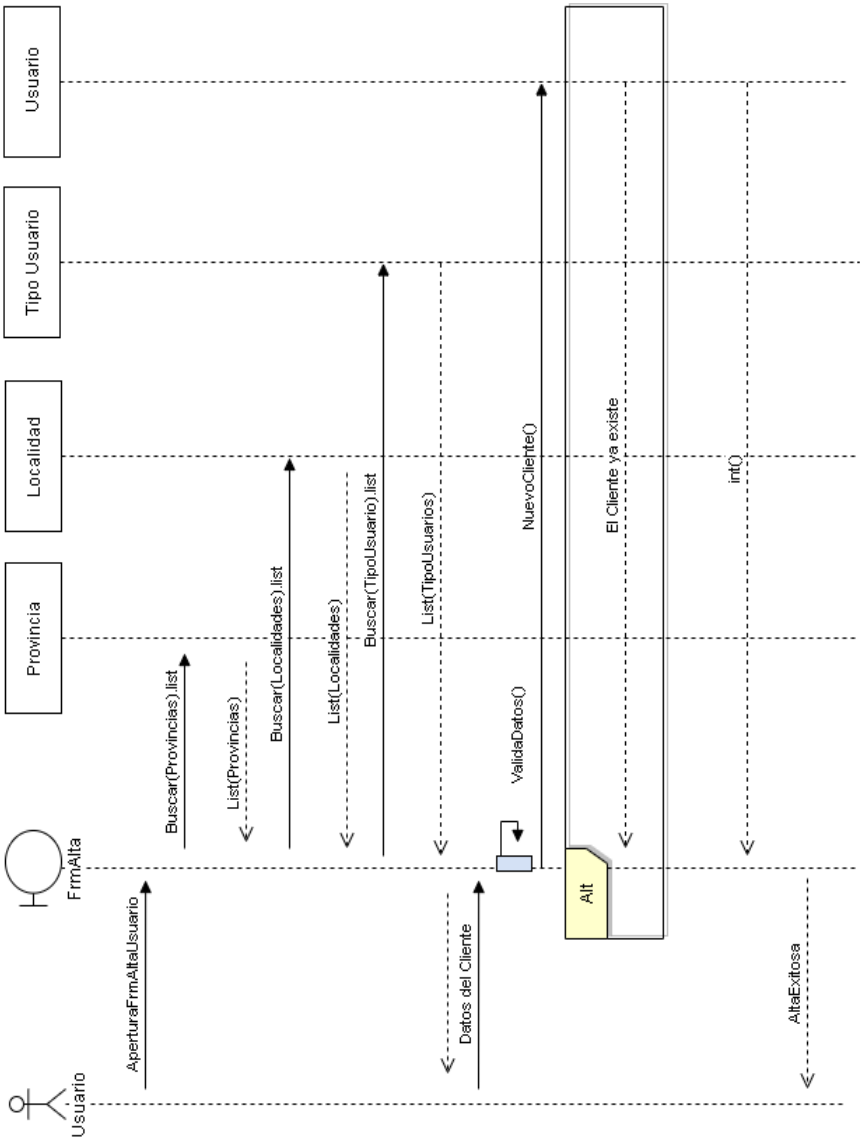

class EcommerceHelper
EcommerceHelper.Entidades::FamiliaEntidad
«property»
+ IdFamilia(): int
+ NombreFamilia(): string
-_Familia
EcommerceHelper.Entidades::PatenteEntidad
«property»
+ Descripcion(): string
+ IdPatente(): int
+_Permisos
EcommerceHelper.Entidades::UsuarioEntidad
- _Familia: FamiliaEntidad = new FamiliaEntidad()
- _Permisos: List<IFamPat> = new List<IFamPat>()
«property»
+ Apellido(): string
+ DVH(): int
+ Email(): string
+ Familia(): FamiliaEntidad
+ IdTipoUsuario(): int
+ IdUsuario(): int
+ MiDireccion(): List<DireccionEntidad >
+ MiDocumento(): TipoDeDocumentoEntidad
+ MiSexo(): SexoEntidad
+ MiTelefono(): TipoTelefonoEntidad
+ MiUsuario(): TipoUsuarioEntidad
+ Nombre(): string
+ NumeroDocumento(): int
+ NumeroTelefono(): Int64
+ Password(): string
+ Permisos(): List<IFamPat>
0..*
1
EcommerceHelper.Entidades::TipoTelefonoEntidad
«property»
+ Descripcion(): string
+ IdTipoTelefono(): int
1
EcommerceHelper.Entidades::SexoEntidad
«property»
+ Descripcion(): string
+ IdSexo(): int
1
1..*
1
EcommerceHelper.Entidades::DireccionEntidad
+ _MiLocalidad: LocalidadEntidad = new LocalidadEn...
+ _MiProvincia: ProvinciaEntidad = new ProvinciaEn...
+ _MiTipoDireccion: TipoDireccionEntidad = new TipoDirecci...
«property»
+ Calle(): string
+ Departamento(): string
+ IdDireccion(): int
+ MiLocalidad(): LocalidadEntidad
+ MiProvincia(): ProvinciaEntidad
+ MiTipoDireccion(): TipoDireccionEntidad
+ Numero(): int
+ Piso(): string
«use»
EcommerceHelper.BLL::UsuarioBLL
- _DalDireccion: DireccionDAL
- _dalProvicia: ProvinciaDAL
- _DalUsuario: UsuarioDAL
+ ActualizarPassword(string, string): UsuarioEntidad
+ BuscarMail(string): UsuarioEntidad
+ EliminarDireccion(int, int): void
+ GenerarCadenaDVHTablaEntera(): List< UsuarioEntidad>
+ IniciarSesion(string, string): UsuarioEntidad
+ InsertDireccionDeFacturacion(DireccionEntidad, UsuarioEntidad): void
+ InsertFamiliaUsuario(int, int, string): int
+ RegistrarUsuario(UsuarioEntidad): int
+ Select(int): UsuarioEntidad
+ SelectALLEmpleadoDeLimpieza(): List<UsuarioEntidad>
+ SelectALLEmpleadoDeLimpiezabyLocalidad(int): List<UsuarioEntidad>
+ SelectALLEmpleados(): List<UsuarioEntidad>
+ SelectALLProvincias(): List<ProvinciaEntidad>
+ SelectALLSexos(): List<SexoEntidad>
+ UpdateDatosEmpleado(UsuarioEntidad): void
+ UpdateDireccionEmpleado(DireccionEntidad, UsuarioEntidad): void
+ UsuarioBLL()
+ UsuarioModificarPermisos(List<IFamPat>, List<IFamPat>, string): bool
+ UsuarioTraerPermisos(string, int): List<IFamPat>
+_MiLocalidad
EcommerceHelper.Entidades::LocalidadEntidad
«property»
+ Descripcion(): string
+ IdLocalidad(): int
+ MiProvincia(): ProvinciaEntidad

Diagrama de Secuencia CU - 007 – Recursos Humanos



Pantalla

Alta Empleado

CU - 008 – Asignación de Perfiles

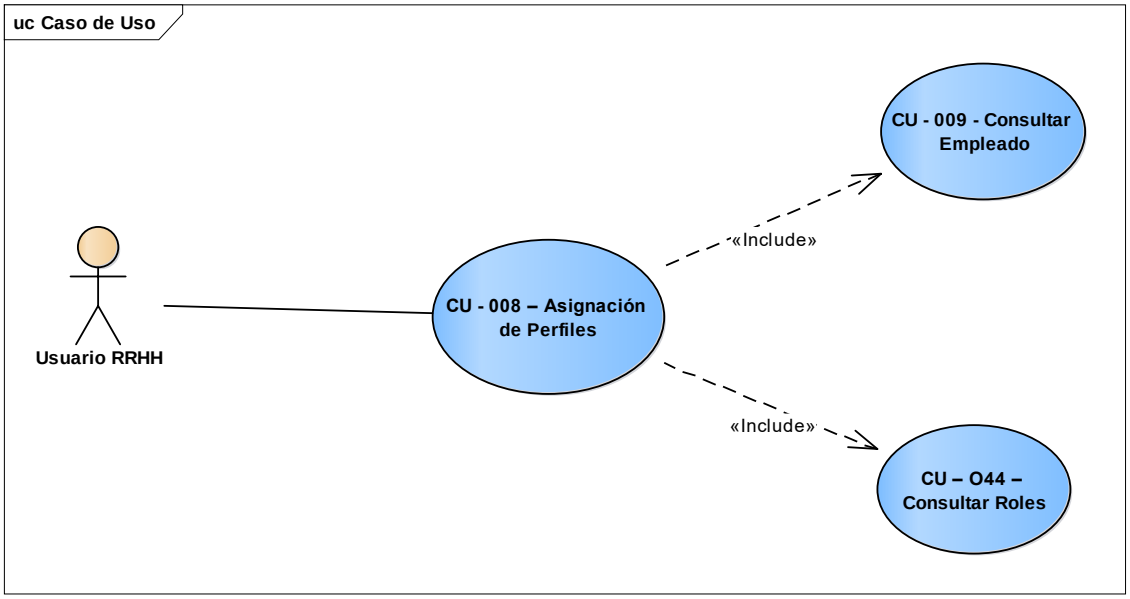
Precondición
El empleado debe existir en el sistema

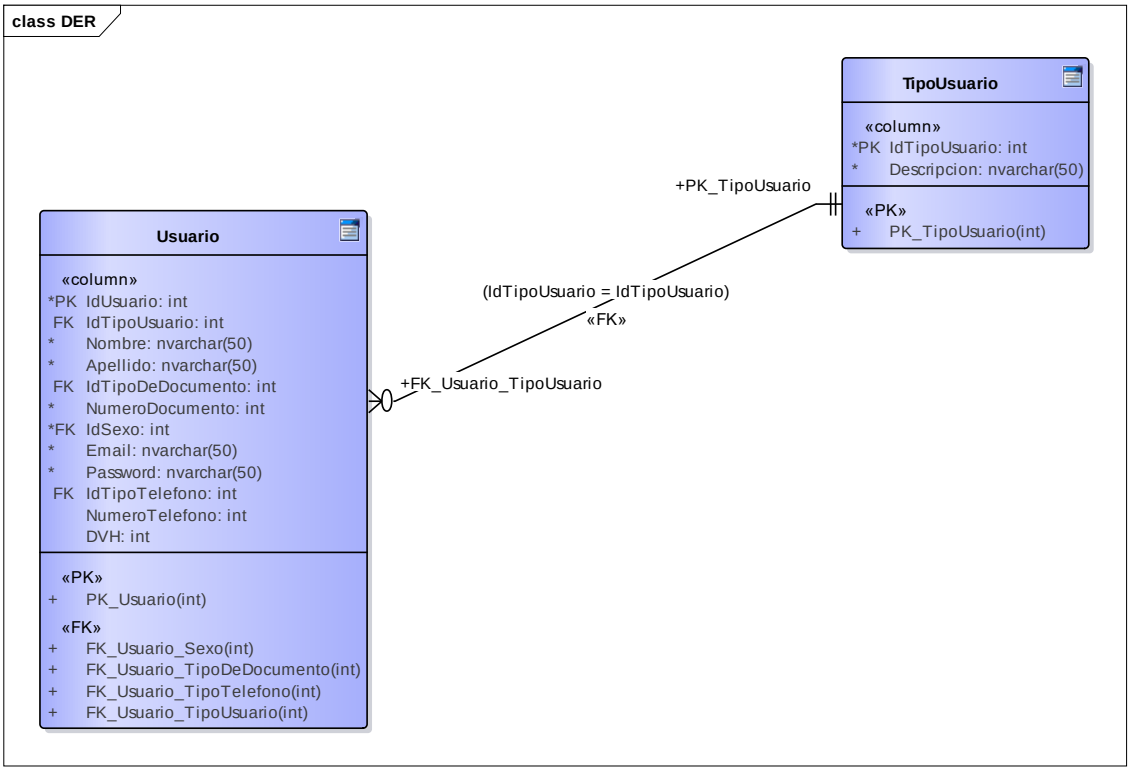
Actores primarios:
RRHH

Camino principal del caso de uso – Asignación de Perfiles

1. El personal de RRHH ingresa a Asignar perfiles.
2. El sistema despliega un listado de los empleados disponibles para asignarle perfil
3. El personal selecciona al empleado.
4. Manteniéndose dentro de la misma pantalla del debajo figuran los posibles perfiles dentro de una lista desplegable.
5. El personal de RRHH selecciona el perfil a asignar al empleado.
6. El sistema registra el nuevo perfil y actualiza la base de datos.

Postcondición
Nuevo perfil asignado al empleado


uc Caso de Uso
CU - 009 - Consultar Empleado
«Include»
Usuario RRHH
CU - 008 – Asignación de Perfiles
«Include»
CU – O44 – Consultar Roles


class DER
TipoUsuario
«column»
*PK IdTipoUsuario: int
* Descripcion: nvarchar(50)
«PK»
+ PK_TipoUsuario(int)
+PK_TipoUsuario
(IdTipoUsuario = IdTipoUsuario)
«FK»
+FK_Usuario_TipoUsuario
Usuario
«column»
*PK IdUsuario: int
FK IdTipoUsuario: int
* Nombre: nvarchar(50)
* Apellido: nvarchar(50)
FK IdTipoDeDocumento: int
* NumeroDocumento: int
*FK IdSexo: int
* Email: nvarchar(50)
* Password: nvarchar(50)
FK IdTipoTelefono: int
NumeroTelefono: int
DVH: int
«PK»
+ PK_Usuario(int)
«FK»
+ FK_Usuario_Sexo(int)
+ FK_Usuario_TipoDeDocumento(int)
+ FK_Usuario_TipoTelefono(int)
+ FK_Usuario_TipoUsuario(int)

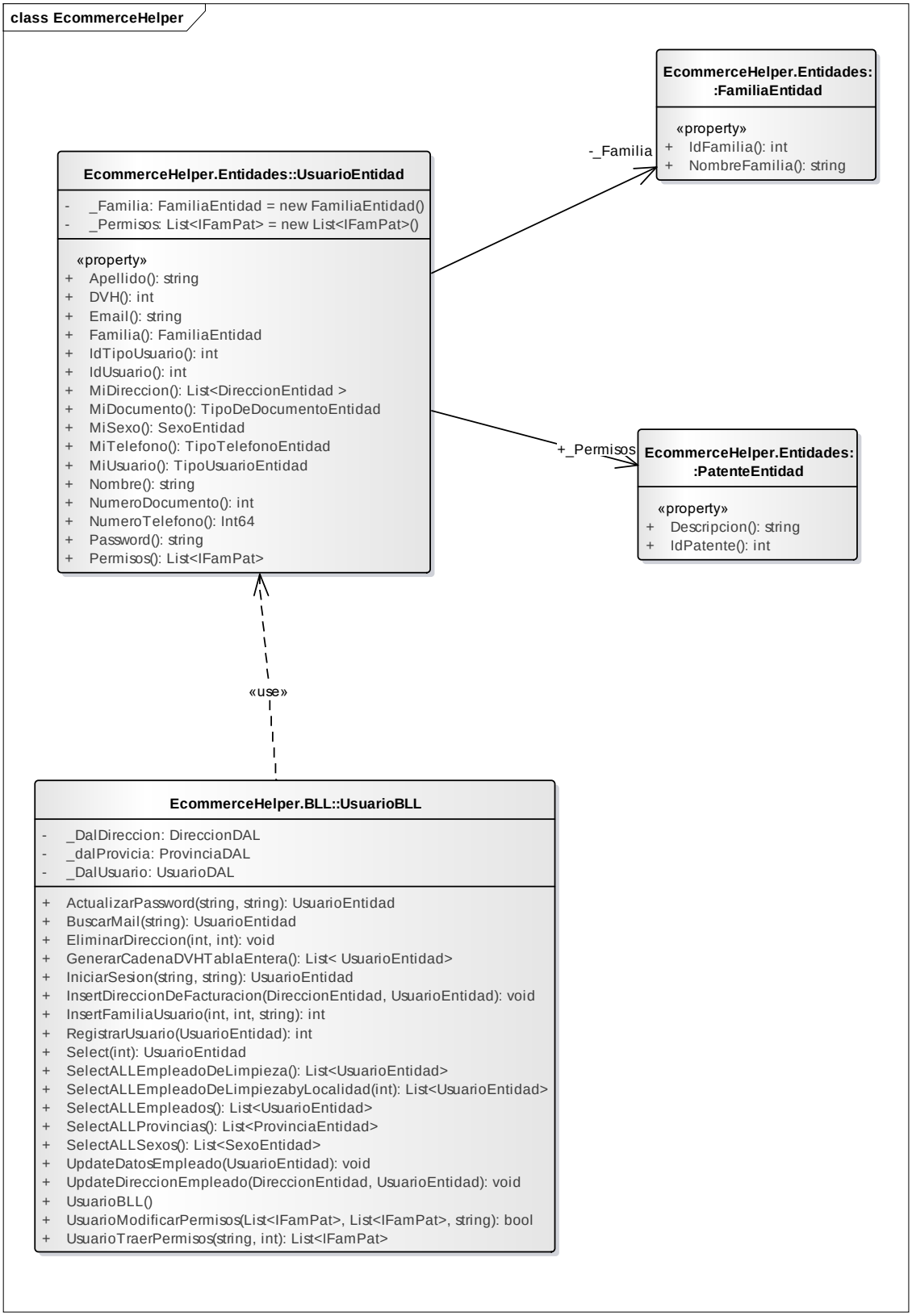
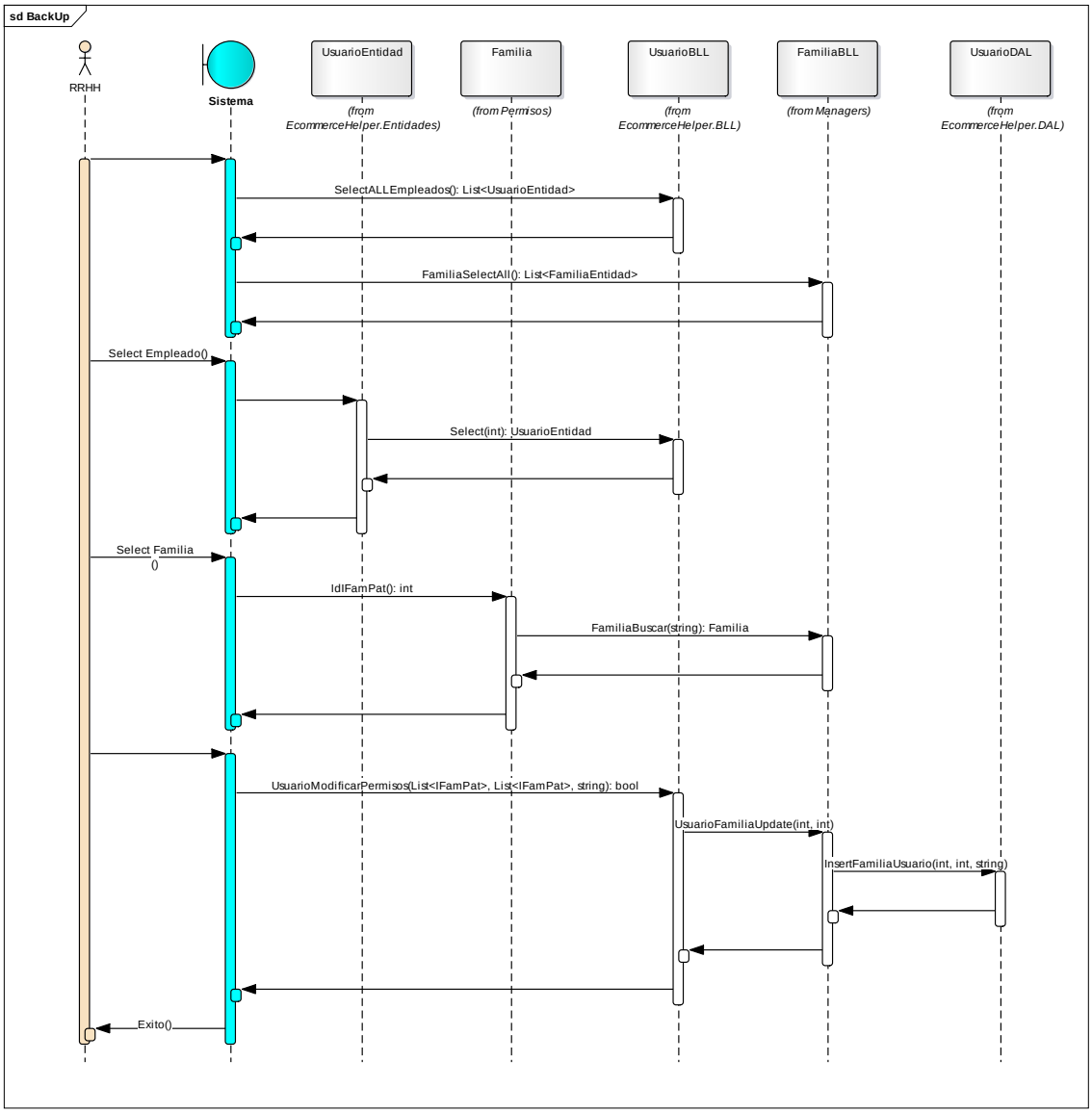

class EcommerceHelper
EcommerceHelper.Entidades::FamiliaEntidad
«property»
+ IdFamilia(): int
+ NombreFamilia(): string
-_Familia
EcommerceHelper.Entidades::UsuarioEntidad
- _Familia: FamiliaEntidad = new FamiliaEntidad()
- _Permisos: List<IFamPat> = new List<IFamPat>()
«property»
+ Apellido(): string
+ DVH(): int
+ Email(): string
+ Familia(): FamiliaEntidad
+ IdTipoUsuario(): int
+ IdUsuario(): int
+ MiDireccion(): List<DireccionEntidad >
+ MiDocumento(): TipoDeDocumentoEntidad
+ MiSexo(): SexoEntidad
+ MiTelefono(): TipoTelefonoEntidad
+ MiUsuario(): TipoUsuarioEntidad
+ Nombre(): string
+ NumeroDocumento(): int
+ NumeroTelefono(): Int64
+ Password(): string
+ Permisos(): List<IFamPat>
+_Permisos
EcommerceHelper.Entidades::PatenteEntidad
«property»
+ Descripcion(): string
+ IdPatente(): int
«use»
EcommerceHelper.BLL::UsuarioBLL
- _DalDireccion: DireccionDAL
- _dalProvicia: ProvinciaDAL
- _DalUsuario: UsuarioDAL
+ ActualizarPassword(string, string): UsuarioEntidad
+ BuscarMail(string): UsuarioEntidad
+ EliminarDireccion(int, int): void
+ GenerarCadenaDVHTablaEntera(): List< UsuarioEntidad>
+ IniciarSesion(string, string): UsuarioEntidad
+ InsertDireccionDeFacturacion(DireccionEntidad, UsuarioEntidad): void
+ InsertFamiliaUsuario(int, int, string): int
+ RegistrarUsuario(UsuarioEntidad): int
+ Select(int): UsuarioEntidad
+ SelectALLEmpleadoDeLimpieza(): List<UsuarioEntidad>
+ SelectALLEmpleadoDeLimpiezabyLocalidad(int): List<UsuarioEntidad>
+ SelectALLEmpleados(): List<UsuarioEntidad>
+ SelectALLProvincias(): List<ProvinciaEntidad>
+ SelectALLSexos(): List<SexoEntidad>
+ UpdateDatosEmpleado(UsuarioEntidad): void
+ UpdateDireccionEmpleado(DireccionEntidad, UsuarioEntidad): void
+ UsuarioBLL()
+ UsuarioModificarPermisos(List<IFamPat>, List<IFamPat>, string): bool
+ UsuarioTraerPermisos(string, int): List<IFamPat>

Diagrama de Secuencia CU - 008 – Asignación de Perfiles



Pantallas

Seleccione Empleado
messiano, leandro
Seleccione Perfil
Publico
Aceptar
Cancelar

CU - 030 – Modificar Empleado

Precondición
El personal de recursos humanos debe estar logueado.

Actores primarios:
Recursos humanos.

Camino principal del caso de uso – Modificar Empleado

1. El usuario de recursos humanos selecciona de la grilla de empleados modificar a un empleado.
2. El sistema carga en el formulario los siguientes datos
    a. E-mail: Texto con formato ejemplo@ejemplo.com
    b. Nombres o Nombre: Texto
    c. Apellidos: Texto.
    d. Sexo
    e. DNI
    f. Tipo de teléfono
    g. Número de teléfono (mínimo 8 dígitos) 1234-5678

    h. Dirección

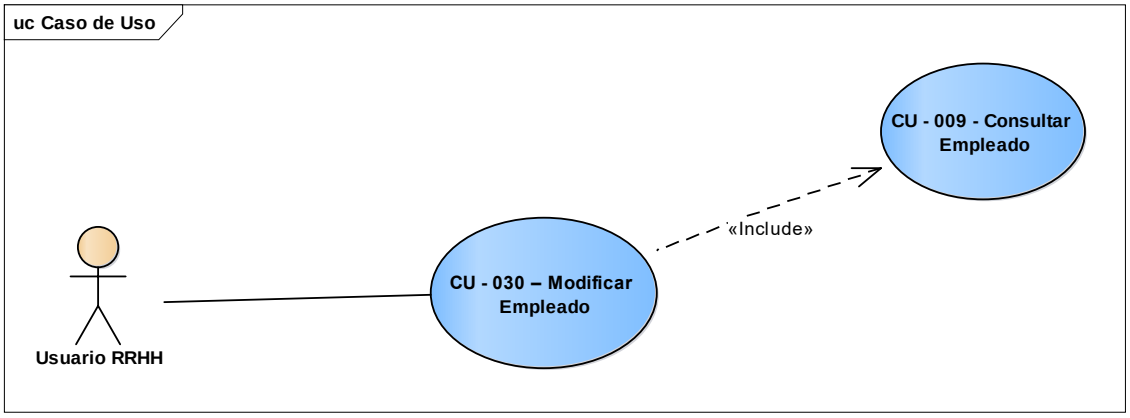
        i. Provincia: texto
        ii. Localidad: texto
        iii. Calle: texto
        iv. Nro. de calle: Números
        v. Piso: Número.
        vi. Departamento: texto
        vii. Tipo de Domicilio

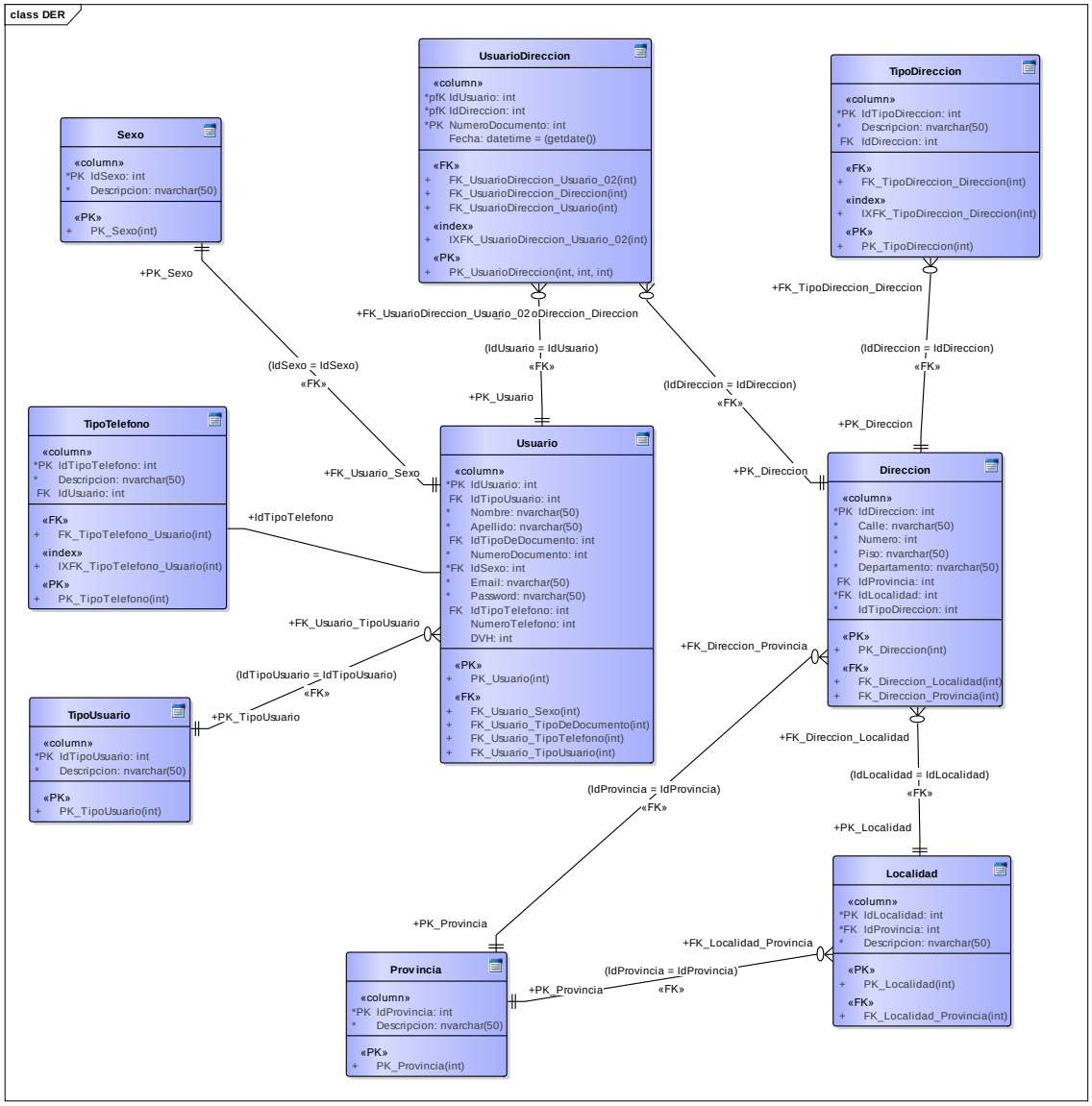
3. El usuario de recursos humanos realiza las modificaciones que desea y confirma la operación de guardar cambios.
4. El sistema valida los datos.
5. El sistema genera el Digito Verificador horizontal.
6. El sistema registra las modificaciones del empleado.
7. El sistema registra en la bitácora el evento de modificación del empleado.
8. El sistema actualiza el valor del Digito Verificador Vertical tabla Usuario.

Postcondición
Las modificaciones se encuentran registradas

Flujo alternativo

Falla en el momento de grabación de las modificaciones del empleado.
El sistema registra en bitácora el error producido.


uc Caso de Uso
CU - 009 - Consultar Empleado
«Include»
CU - 030 – Modificar Empleado
Usuario RRHH


class DER
Sexo
«column»
*PK IdSexo: int
* Descripcion: nvarchar(50)
«PK»
+ PK_Sexo(int)
UsuarioDireccion
«column»
*pfK IdUsuario: int
*pfK IdDireccion: int
*PK NumeroDocumento: int
Fecha: datetime = (getdate())
«FK»
+ FK_UsuarioDireccion_Usuario_02(int)
+ FK_UsuarioDireccion_Direccion(int)
+ FK_UsuarioDireccion_Usuario(int)
«index»
+ IXFK_UsuarioDireccion_Usuario_02(int)
«PK»
+ PK_UsuarioDireccion(int, int, int)
TipoDireccion
«column»
*PK IdTipoDireccion: int
* Descripcion: nvarchar(50)
FK IdDireccion: int
«FK»
+ FK_TipoDireccion_Direccion(int)
«index»
+ IXFK_TipoDireccion_Direccion(int)
«PK»
+ PK_TipoDireccion(int)
+PK_Sexo
+FK_UsuarioDireccion_Usuario_02
+FK_TipoDireccion_Direccion
(IdSexo = IdSexo)
«FK»
(IdUsuario = IdUsuario)
«FK»
(IdDireccion = IdDireccion)
«FK»
(IdDireccion = IdDireccion)
«FK»
+PK_Usuario
+PK_Direccion
TipoTelefono
«column»
*PK IdTipoTelefono: int
* Descripcion: nvarchar(50)
FK IdUsuario: int
«FK»
+ FK_TipoTelefono_Usuario(int)
«index»
+ IXFK_TipoTelefono_Usuario(int)
«PK»
+ PK_TipoTelefono(int)
+FK_Usuario_Sexo
+PK_Direccion
+IdTipoTelefono
Usuario
«column»
*PK IdUsuario: int
FK IdTipoUsuario: int
* Nombre: nvarchar(50)
* Apellido: nvarchar(50)
FK IdTipoDeDocumento: int
* NumeroDocumento: int
*FK IdSexo: int
* Email: nvarchar(50)
* Password: nvarchar(50)
FK IdTipoTelefono: int
NumeroTelefono: int
DVH: int
«PK»
+ PK_Usuario(int)
«FK»
+ FK_Usuario_Sexo(int)
+ FK_Usuario_TipoDeDocumento(int)
+ FK_Usuario_TipoTelefono(int)
+ FK_Usuario_TipoUsuario(int)
Direccion
«column»
*PK IdDireccion: int
* Calle: nvarchar(50)
* Numero: int
* Piso: nvarchar(50)
* Departamento: nvarchar(50)
FK IdProvincia: int
*FK IdLocalidad: int
* IdTipoDireccion: int
«PK»
+ PK_Direccion(int)
«FK»
+ FK_Direccion_Localidad(int)
+ FK_Direccion_Provincia(int)
+FK_Usuario_TipoUsuario
+FK_Direccion_Provincia
(IdTipoUsuario = IdTipoUsuario)
«FK»
TipoUsuario
«column»
*PK IdTipoUsuario: int
* Descripcion: nvarchar(50)
«PK»
+ PK_TipoUsuario(int)
+PK_TipoUsuario
+FK_Direccion_Localidad
(IdProvincia = IdProvincia)
«FK»
(IdLocalidad = IdLocalidad)
«FK»
+PK_Localidad
Localidad
«column»
*PK IdLocalidad: int
*FK IdProvincia: int
* Descripcion: nvarchar(50)
«PK»
+ PK_Localidad(int)
«FK»
+ FK_Localidad_Provincia(int)
+PK_Provincia
+FK_Localidad_Provincia
(IdProvincia = IdProvincia)
«FK»
+PK_Provincia
Provincia
«column»
*PK IdProvincia: int
* Descripcion: nvarchar(50)
«PK»
+ PK_Provincia(int)

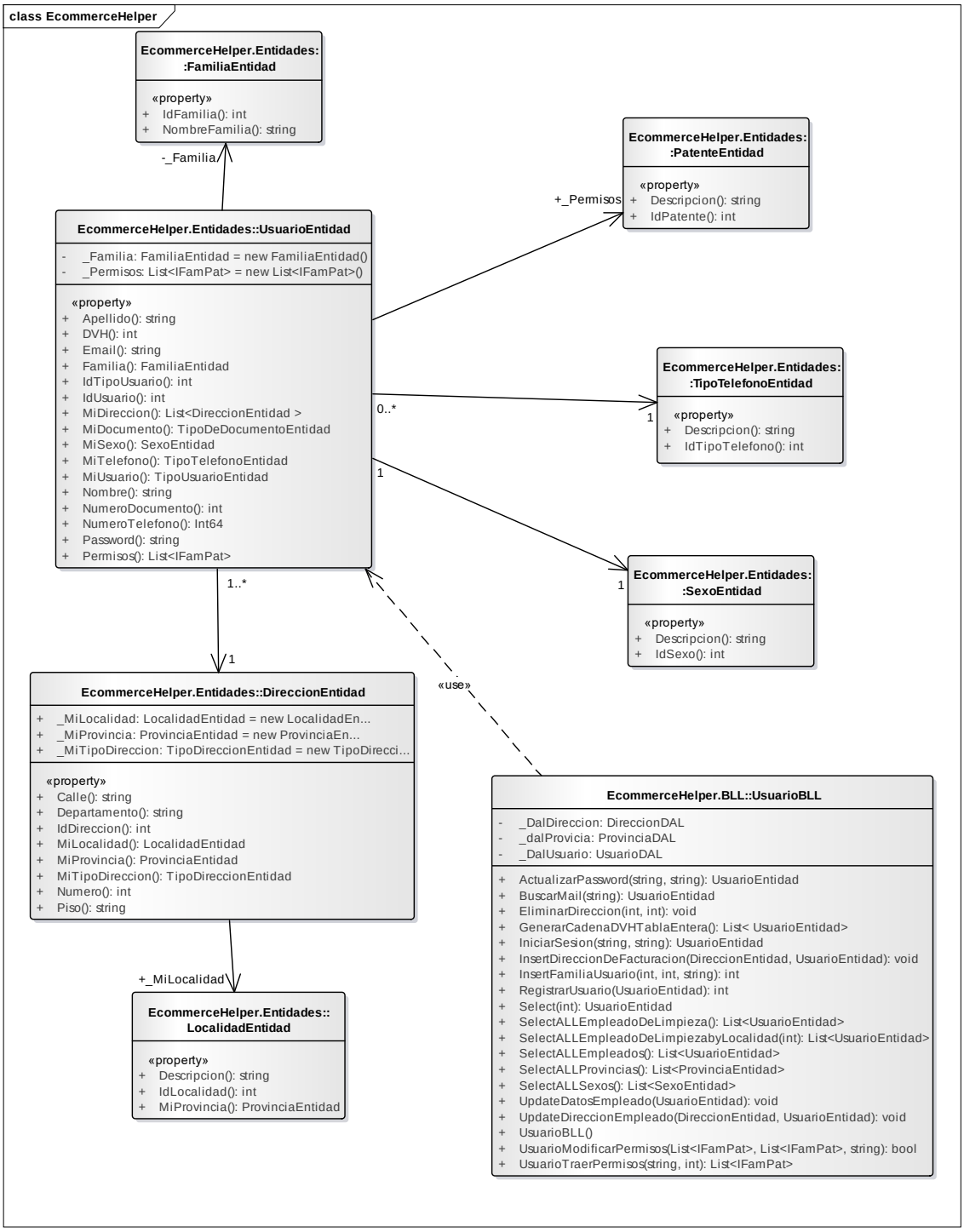
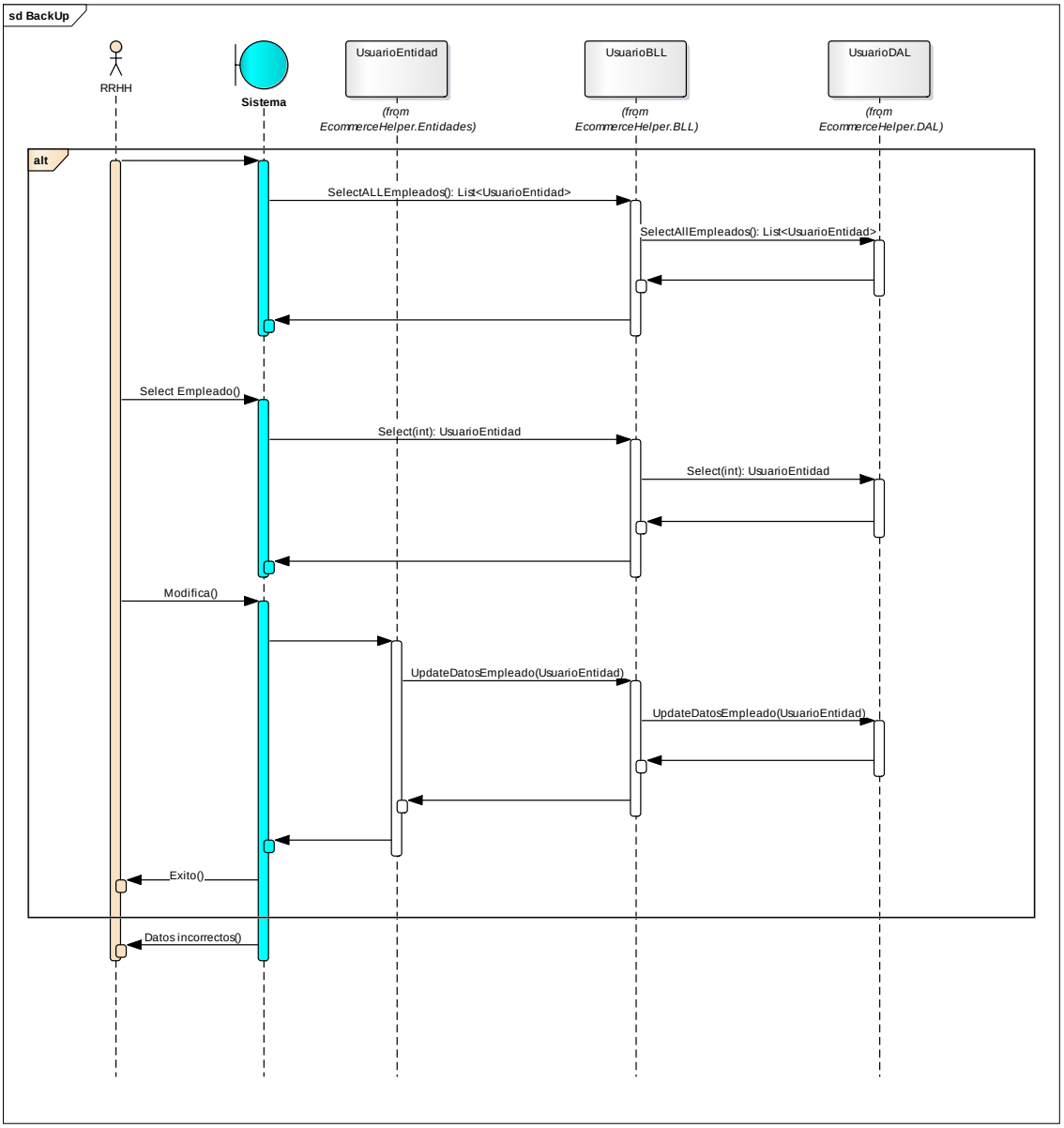

class EcommerceHelper
EcommerceHelper.Entidades::FamiliaEntidad
«property»
+ IdFamilia(): int
+ NombreFamilia(): string
-_Familia
EcommerceHelper.Entidades::PatenteEntidad
«property»
+ Descripcion(): string
+ IdPatente(): int
+_Permisos
EcommerceHelper.Entidades::UsuarioEntidad
- _Familia: FamiliaEntidad = new FamiliaEntidad()
- _Permisos: List<IFamPat> = new List<IFamPat>()
«property»
+ Apellido(): string
+ DVH(): int
+ Email(): string
+ Familia(): FamiliaEntidad
+ IdTipoUsuario(): int
+ IdUsuario(): int
+ MiDireccion(): List<DireccionEntidad >
+ MiDocumento(): TipoDeDocumentoEntidad
+ MiSexo(): SexoEntidad
+ MiTelefono(): TipoTelefonoEntidad
+ MiUsuario(): TipoUsuarioEntidad
+ Nombre(): string
+ NumeroDocumento(): int
+ NumeroTelefono(): Int64
+ Password(): string
+ Permisos(): List<IFamPat>
0..*
1
EcommerceHelper.Entidades::TipoTelefonoEntidad
«property»
+ Descripcion(): string
+ IdTipoTelefono(): int
1
EcommerceHelper.Entidades::SexoEntidad
«property»
+ Descripcion(): string
+ IdSexo(): int
1
1..*
1
EcommerceHelper.Entidades::DireccionEntidad
+ _MiLocalidad: LocalidadEntidad = new LocalidadEn...
+ _MiProvincia: ProvinciaEntidad = new ProvinciaEn...
+ _MiTipoDireccion: TipoDireccionEntidad = new TipoDirecci...
«property»
+ Calle(): string
+ Departamento(): string
+ IdDireccion(): int
+ MiLocalidad(): LocalidadEntidad
+ MiProvincia(): ProvinciaEntidad
+ MiTipoDireccion(): TipoDireccionEntidad
+ Numero(): int
+ Piso(): string
«use»
EcommerceHelper.BLL::UsuarioBLL
- _DalDireccion: DireccionDAL
- _dalProvicia: ProvinciaDAL
- _DalUsuario: UsuarioDAL
+ ActualizarPassword(string, string): UsuarioEntidad
+ BuscarMail(string): UsuarioEntidad
+ EliminarDireccion(int, int): void
+ GenerarCadenaDVHTablaEntera(): List< UsuarioEntidad>
+ IniciarSesion(string, string): UsuarioEntidad
+ InsertDireccionDeFacturacion(DireccionEntidad, UsuarioEntidad): void
+ InsertFamiliaUsuario(int, int, string): int
+ RegistrarUsuario(UsuarioEntidad): int
+ Select(int): UsuarioEntidad
+ SelectALLEmpleadoDeLimpieza(): List<UsuarioEntidad>
+ SelectALLEmpleadoDeLimpiezabyLocalidad(int): List<UsuarioEntidad>
+ SelectALLEmpleados(): List<UsuarioEntidad>
+ SelectALLProvincias(): List<ProvinciaEntidad>
+ SelectALLSexos(): List<SexoEntidad>
+ UpdateDatosEmpleado(UsuarioEntidad): void
+ UpdateDireccionEmpleado(DireccionEntidad, UsuarioEntidad): void
+ UsuarioBLL()
+ UsuarioModificarPermisos(List<IFamPat>, List<IFamPat>, string): bool
+ UsuarioTraerPermisos(string, int): List<IFamPat>
+_MiLocalidad
EcommerceHelper.Entidades::LocalidadEntidad
«property»
+ Descripcion(): string
+ IdLocalidad(): int
+ MiProvincia(): ProvinciaEntidad

Diagrama de Secuencia CU - 030 – Modificar Empleado



Pantalla

Listado de Empleados

Nombre	Apellido	NumeroDocumento	NumeroTelefono	IdSexo	Email	IdDireccion	Calle	Numero	Piso	Departamento	IdLocalidad	IdProvincia		
leandro	messiano	33897289	237465898	2	messi3@gmail.com	16	avellaneda	876	0	0	4	1	Modificar	Eliminar
Susana	Alegria	23878111	237477188	1	alegriasusana@gmail.com	37	escueles	234	0	a	4	1	Modificar	Eliminar
leandro	messiano	33897289	237465898	2	messi3@gmail.com	38	Florencio Sánchez	1300	0	0	2	1	Modificar	Eliminar
Pablo Martin	Strasser	27666555	237484998	2	PabloMartin@gmail.com	40	Int. Luis Tulissi	1930	0	0	3	1	Modificar	Eliminar

CU - 031 – Eliminar Empleado

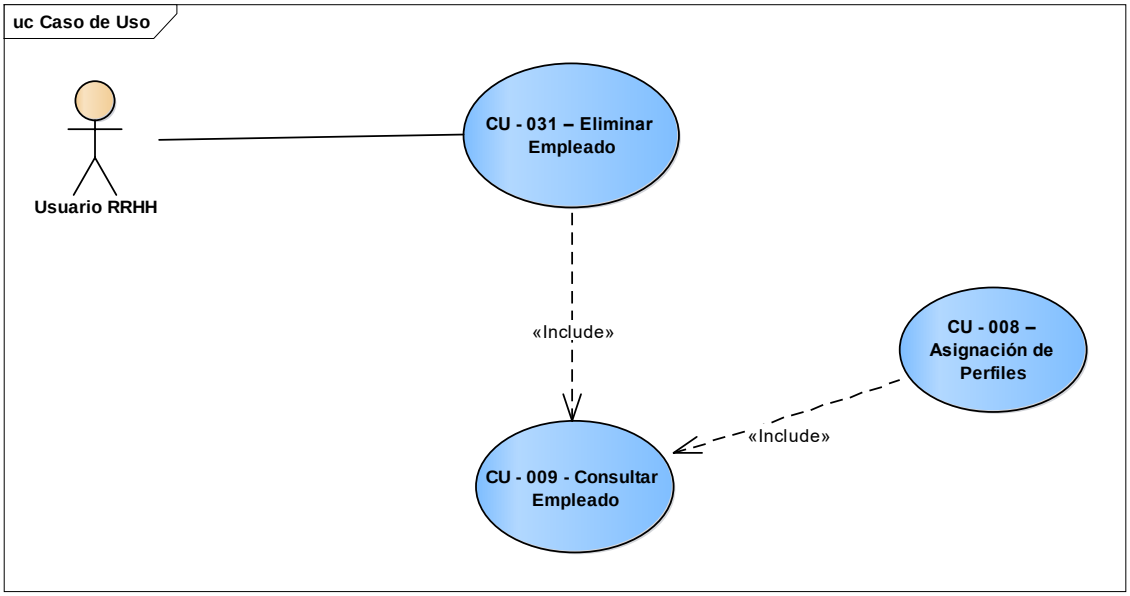
Precondición
El usuario de recursos humanos debe estar logueado con los permisos correspondientes para poder acceder al dar de baja de un empleado.

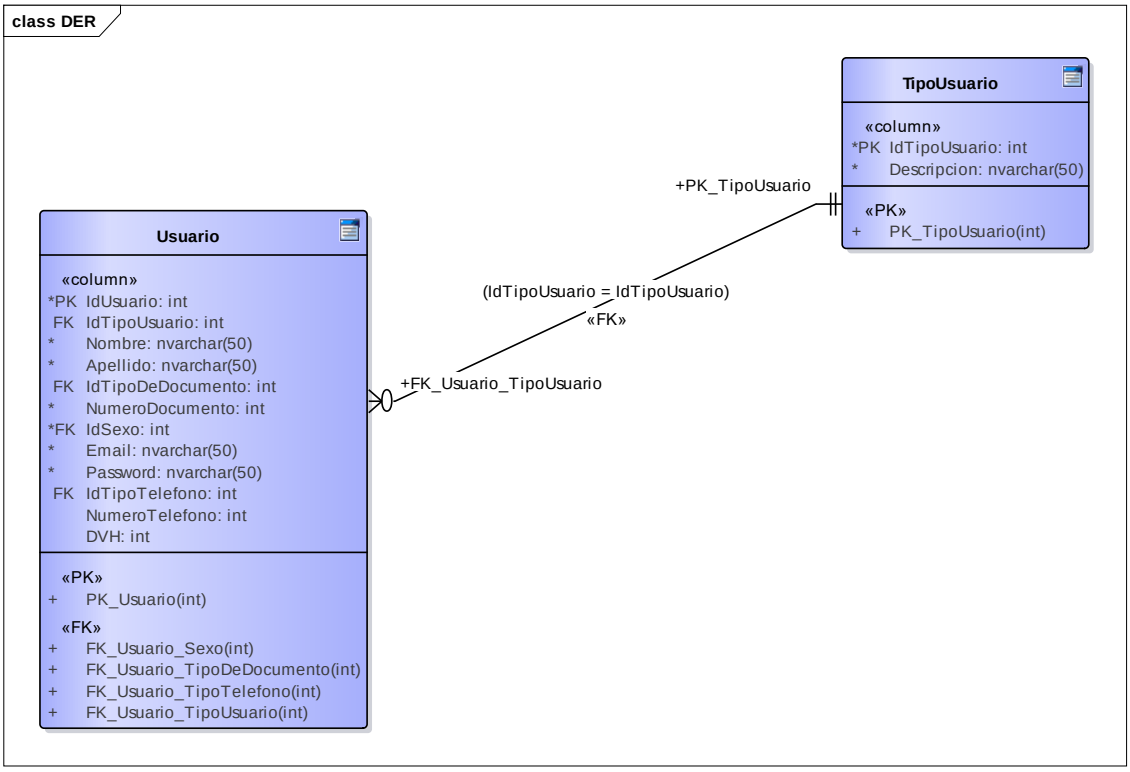
Actores primarios:
Usuario Cliente

Camino principal del caso de uso – Eliminar Empleado

1. l usuario de recursos humanos selecciona de la grilla de empleados el que desea eliminar.
2. El sistema le cambia el empleado su perfil a Publico.
3. El sistema refresca la grilla y actualiza el listado de empleados.

Postcondición
El empleado se encuentra eliminado del perfil empleado.


uc Caso de Uso
Usuario RRHH
CU - 031 – Eliminar Empleado
«Include»
CU - 008 – Asignación de Perfiles
«Include»
CU - 009 - Consultar Empleado


class DER
TipoUsuario
«column»
*PK IdTipoUsuario: int
* Descripcion: nvarchar(50)
«PK»
+ PK_TipoUsuario(int)
+PK_TipoUsuario
(IdTipoUsuario = IdTipoUsuario)
«FK»
+FK_Usuario_TipoUsuario
Usuario
«column»
*PK IdUsuario: int
FK IdTipoUsuario: int
* Nombre: nvarchar(50)
* Apellido: nvarchar(50)
FK IdTipoDeDocumento: int
* NumeroDocumento: int
*FK IdSexo: int
* Email: nvarchar(50)
* Password: nvarchar(50)
FK IdTipoTelefono: int
NumeroTelefono: int
DVH: int
«PK»
+ PK_Usuario(int)
«FK»
+ FK_Usuario_Sexo(int)
+ FK_Usuario_TipoDeDocumento(int)
+ FK_Usuario_TipoTelefono(int)
+ FK_Usuario_TipoUsuario(int)

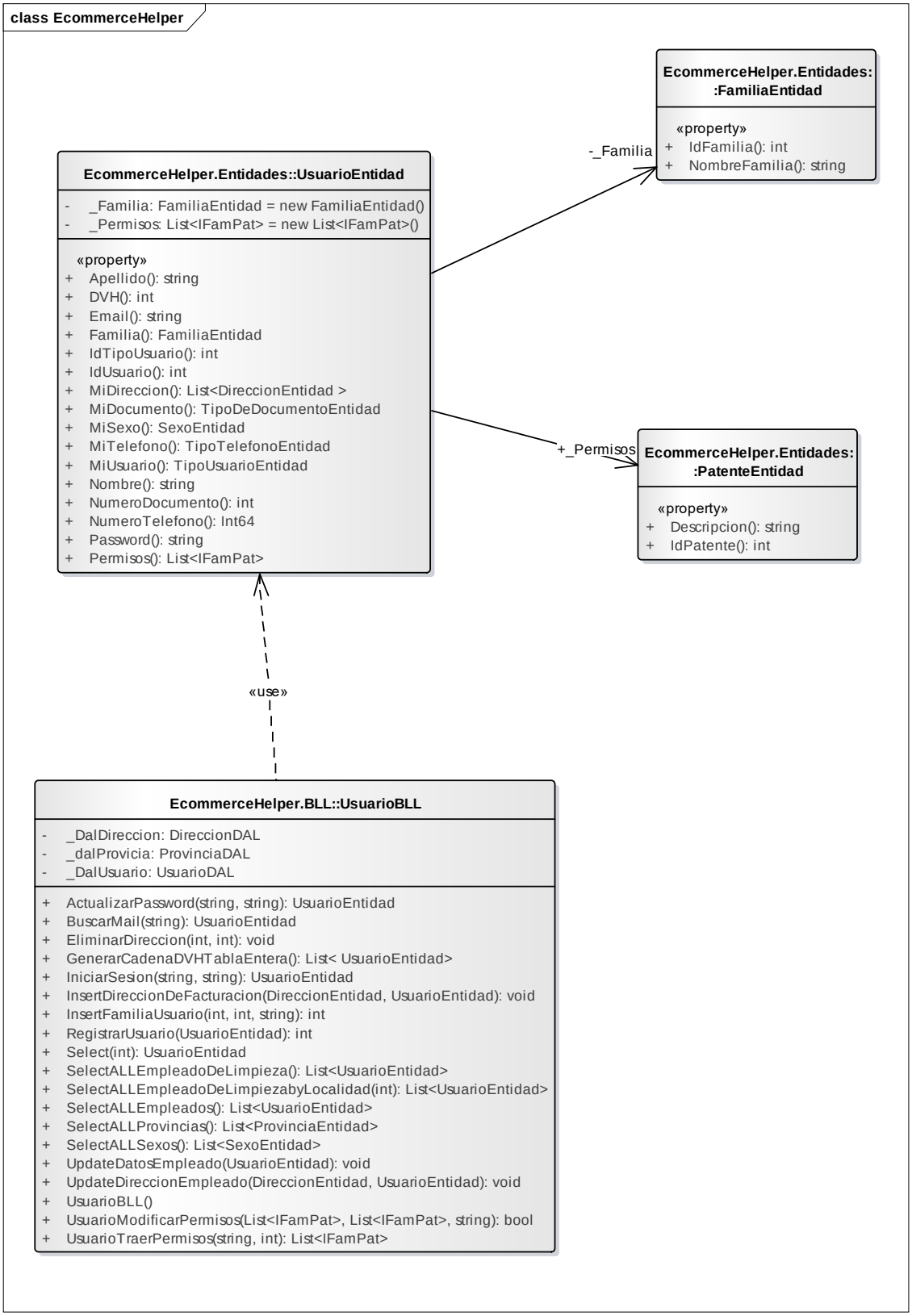
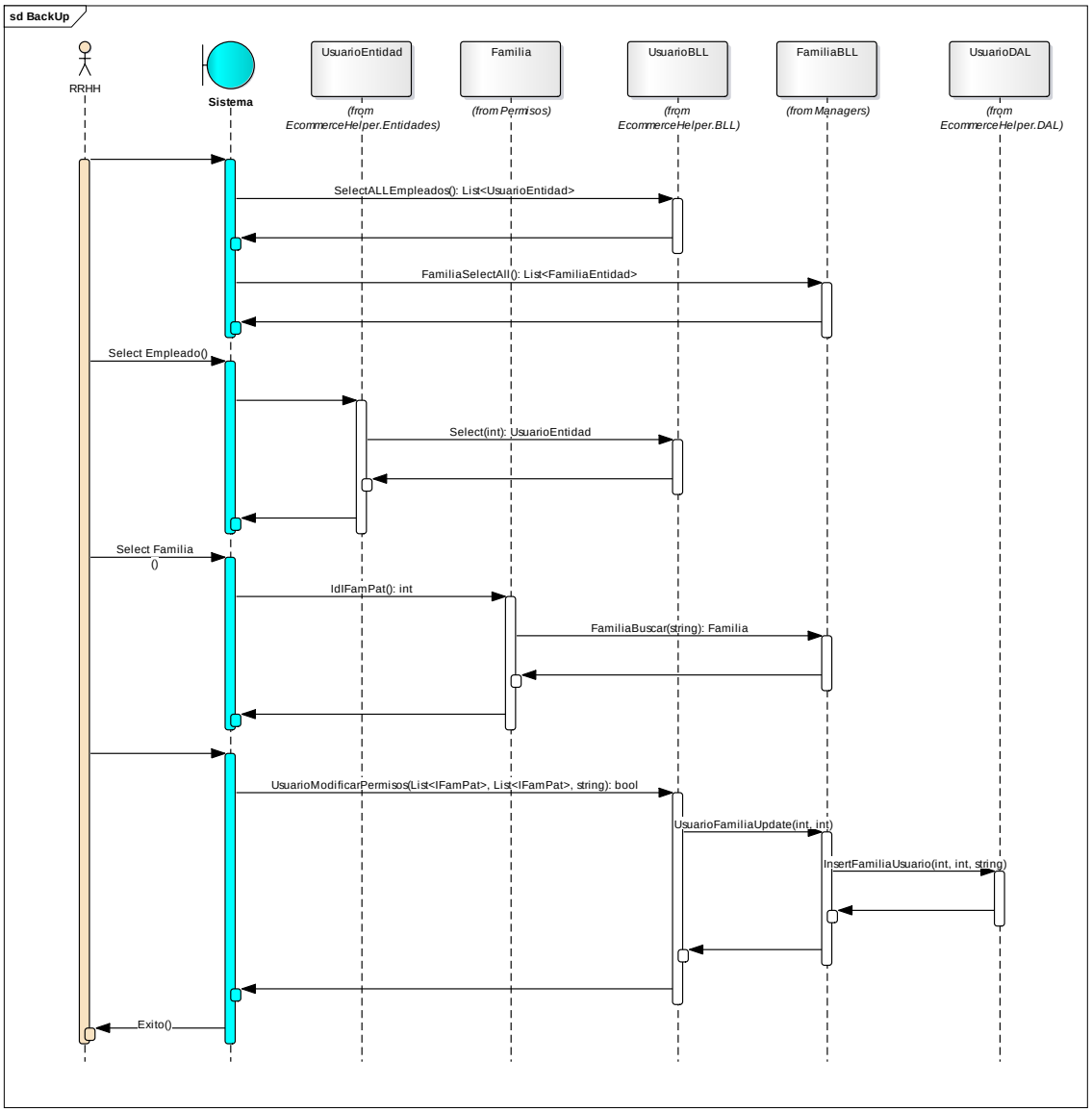

class EcommerceHelper
EcommerceHelper.Entidades::FamiliaEntidad
«property»
+ IdFamilia(): int
+ NombreFamilia(): string
-_Familia
EcommerceHelper.Entidades::UsuarioEntidad
- _Familia: FamiliaEntidad = new FamiliaEntidad()
- _Permisos: List<IFamPat> = new List<IFamPat>()
«property»
+ Apellido(): string
+ DVH(): int
+ Email(): string
+ Familia(): FamiliaEntidad
+ IdTipoUsuario(): int
+ IdUsuario(): int
+ MiDireccion(): List<DireccionEntidad >
+ MiDocumento(): TipoDeDocumentoEntidad
+ MiSexo(): SexoEntidad
+ MiTelefono(): TipoTelefonoEntidad
+ MiUsuario(): TipoUsuarioEntidad
+ Nombre(): string
+ NumeroDocumento(): int
+ NumeroTelefono(): Int64
+ Password(): string
+ Permisos(): List<IFamPat>
+_Permisos
EcommerceHelper.Entidades::PatenteEntidad
«property»
+ Descripcion(): string
+ IdPatente(): int
«use»
EcommerceHelper.BLL::UsuarioBLL
- _DalDireccion: DireccionDAL
- _dalProvicia: ProvinciaDAL
- _DalUsuario: UsuarioDAL
+ ActualizarPassword(string, string): UsuarioEntidad
+ BuscarMail(string): UsuarioEntidad
+ EliminarDireccion(int, int): void
+ GenerarCadenaDVHTablaEntera(): List< UsuarioEntidad>
+ IniciarSesion(string, string): UsuarioEntidad
+ InsertDireccionDeFacturacion(DireccionEntidad, UsuarioEntidad): void
+ InsertFamiliaUsuario(int, int, string): int
+ RegistrarUsuario(UsuarioEntidad): int
+ Select(int): UsuarioEntidad
+ SelectALLEmpleadoDeLimpieza(): List<UsuarioEntidad>
+ SelectALLEmpleadoDeLimpiezabyLocalidad(int): List<UsuarioEntidad>
+ SelectALLEmpleados(): List<UsuarioEntidad>
+ SelectALLProvincias(): List<ProvinciaEntidad>
+ SelectALLSexos(): List<SexoEntidad>
+ UpdateDatosEmpleado(UsuarioEntidad): void
+ UpdateDireccionEmpleado(DireccionEntidad, UsuarioEntidad): void
+ UsuarioBLL()
+ UsuarioModificarPermisos(List<IFamPat>, List<IFamPat>, string): bool
+ UsuarioTraerPermisos(string, int): List<IFamPat>

Diagrama de Secuencia CU - 031 – Eliminar Empleado



Pantalla

Listado de Empleados

Nombre	Apellido	NumeroDocumento	NumeroTelefono	IdSexo	Email	IdDireccion	Calle	Numero	Piso	Departamento	IdLocalidad	IdProvincia		
leandro	messiano	33897289	237465898	2	messi3@gmail.com	16	avellaneda	876	0	0	4	1	Modificar	Eliminar
Susana	Alegria	23878111	237477188	1	alegriasusana@gmail.com	37	escueles	234	0	a	4	1	Modificar	Eliminar
leandro	messiano	33897289	237465898	2	messi3@gmail.com	38	Florencio Sánchez	1300	0	0	2	1	Modificar	Eliminar
Pablo Martin	Strasser	27666555	237484998	2	PabloMartin@gmail.com	40	Int. Luis Tulissi	1930	0	0	3	1	Modificar	Eliminar

CU - 009 – Consultar Empleado

Precondición
El personal de recursos humanos debe estar logueado.

Actores primarios:
Recursos humanos.

Camino principal del caso de uso – Consultar Empleado

1. El sistema lista los empleados.
2. El usuario de recursos humanos selecciona de la grilla de empleados seleccionar a un empleado.
3. El sistema carga en el formulario los siguientes datos
    a. E-mail: Texto con formato ejemplo@ejemplo.com
    b. Nombres o Nombre: Texto
    c. Apellidos: Texto.
    d. Sexo
    e. DNI
    f. Tipo de teléfono
    g. Número de teléfono (mínimo 8 dígitos) 1234-5678

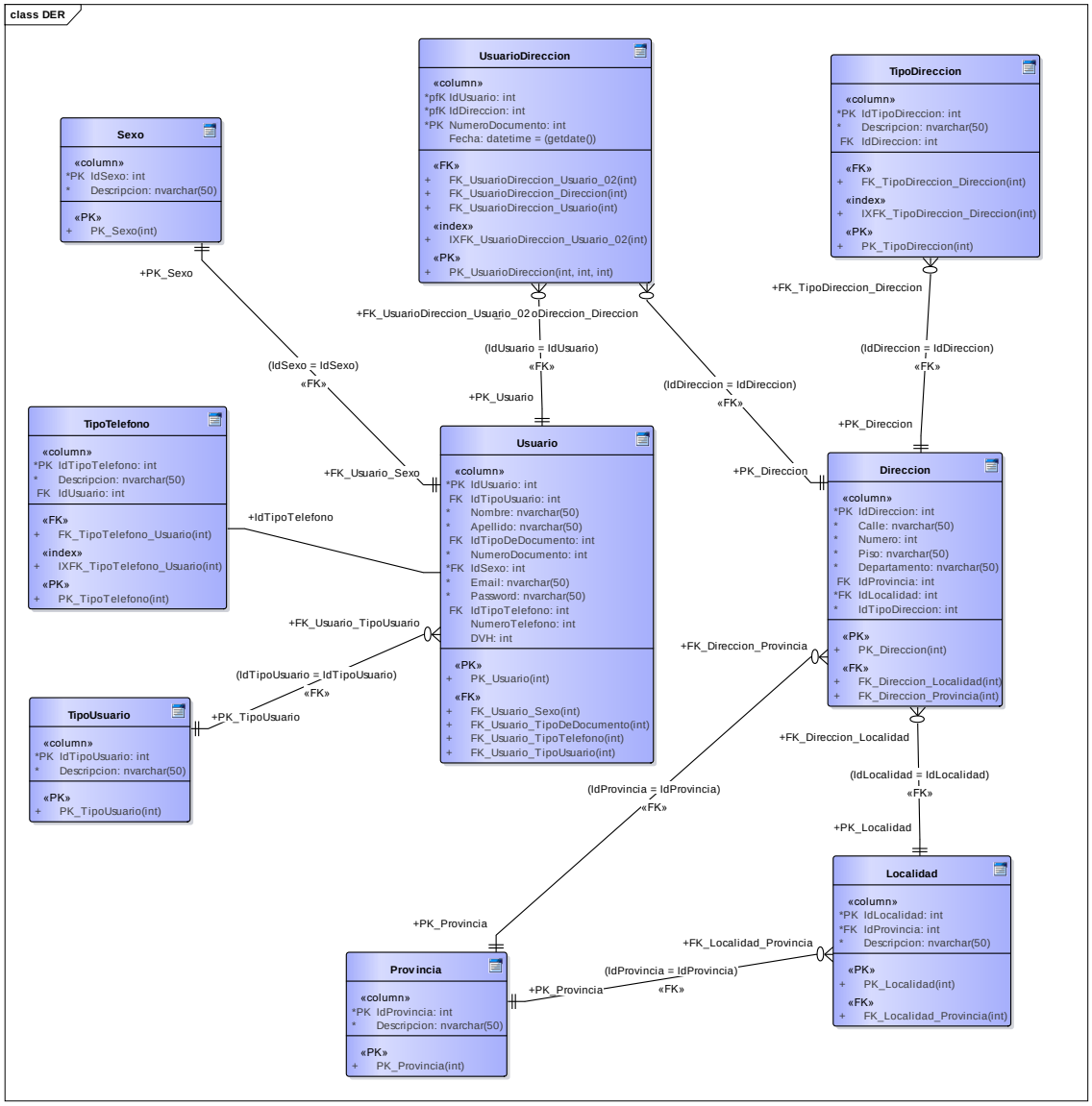
    h. Dirección

        i. Provincia: texto
        ii. Localidad: texto
        iii. Calle: texto
        iv. Nro. de calle: Números
        v. Piso: Número.
        vi. Departamento: texto
        vii. Tipo de Domicilio

Flujo alternativo

No se encuentran empleados registrados.

uc Caso de Uso
Usuario RRHH
CU - 009 - Consultar Empleado


class DER
Sexo
«column»
*PK IdSexo: int
* Descripcion: nvarchar(50)
«PK»
+ PK_Sexo(int)
UsuarioDireccion
«column»
*pfK IdUsuario: int
*pfK IdDireccion: int
*PK NumeroDocumento: int
Fecha: datetime = (getdate())
«FK»
+ FK_UsuarioDireccion_Usuario_02(int)
+ FK_UsuarioDireccion_Direccion(int)
+ FK_UsuarioDireccion_Usuario(int)
«index»
+ IXFK_UsuarioDireccion_Usuario_02(int)
«PK»
+ PK_UsuarioDireccion(int, int, int)
TipoDireccion
«column»
*PK IdTipoDireccion: int
* Descripcion: nvarchar(50)
FK IdDireccion: int
«FK»
+ FK_TipoDireccion_Direccion(int)
«index»
+ IXFK_TipoDireccion_Direccion(int)
«PK»
+ PK_TipoDireccion(int)
+PK_Sexo
+FK_UsuarioDireccion_Usuario_02
+FK_TipoDireccion_Direccion
(IdSexo = IdSexo)
«FK»
(IdUsuario = IdUsuario)
«FK»
(IdDireccion = IdDireccion)
«FK»
(IdDireccion = IdDireccion)
«FK»
+PK_Usuario
+PK_Direccion
TipoTelefono
«column»
*PK IdTipoTelefono: int
* Descripcion: nvarchar(50)
FK IdUsuario: int
«FK»
+ FK_TipoTelefono_Usuario(int)
«index»
+ IXFK_TipoTelefono_Usuario(int)
«PK»
+ PK_TipoTelefono(int)
+FK_Usuario_Sexo
+PK_Direccion
+IdTipoTelefono
Usuario
«column»
*PK IdUsuario: int
FK IdTipoUsuario: int
* Nombre: nvarchar(50)
* Apellido: nvarchar(50)
FK IdTipoDeDocumento: int
* NumeroDocumento: int
*FK IdSexo: int
* Email: nvarchar(50)
* Password: nvarchar(50)
FK IdTipoTelefono: int
NumeroTelefono: int
DVH: int
«PK»
+ PK_Usuario(int)
«FK»
+ FK_Usuario_Sexo(int)
+ FK_Usuario_TipoDeDocumento(int)
+ FK_Usuario_TipoTelefono(int)
+ FK_Usuario_TipoUsuario(int)
Direccion
«column»
*PK IdDireccion: int
* Calle: nvarchar(50)
* Numero: int
* Piso: nvarchar(50)
* Departamento: nvarchar(50)
FK IdProvincia: int
*FK IdLocalidad: int
* IdTipoDireccion: int
«PK»
+ PK_Direccion(int)
«FK»
+ FK_Direccion_Localidad(int)
+ FK_Direccion_Provincia(int)
+FK_Usuario_TipoUsuario
+FK_Direccion_Provincia
(IdTipoUsuario = IdTipoUsuario)
«FK»
TipoUsuario
«column»
*PK IdTipoUsuario: int
* Descripcion: nvarchar(50)
«PK»
+ PK_TipoUsuario(int)
+PK_TipoUsuario
+FK_Direccion_Localidad
(IdProvincia = IdProvincia)
«FK»
(IdLocalidad = IdLocalidad)
«FK»
+PK_Localidad
Localidad
«column»
*PK IdLocalidad: int
*FK IdProvincia: int
* Descripcion: nvarchar(50)
«PK»
+ PK_Localidad(int)
«FK»
+ FK_Localidad_Provincia(int)
+PK_Provincia
+FK_Localidad_Provincia
(IdProvincia = IdProvincia)
«FK»
+PK_Provincia
Provincia
«column»
*PK IdProvincia: int
* Descripcion: nvarchar(50)
«PK»
+ PK_Provincia(int)

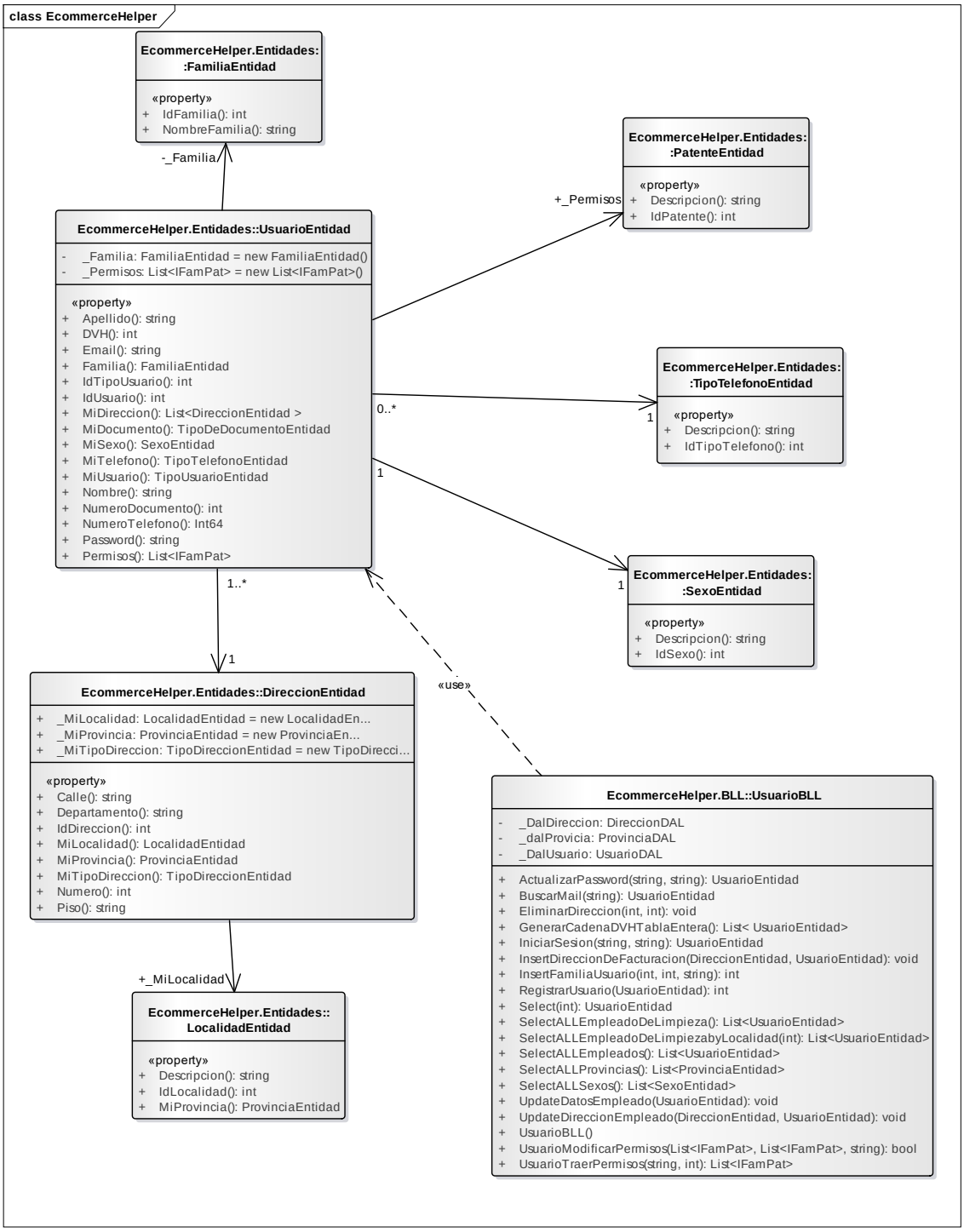
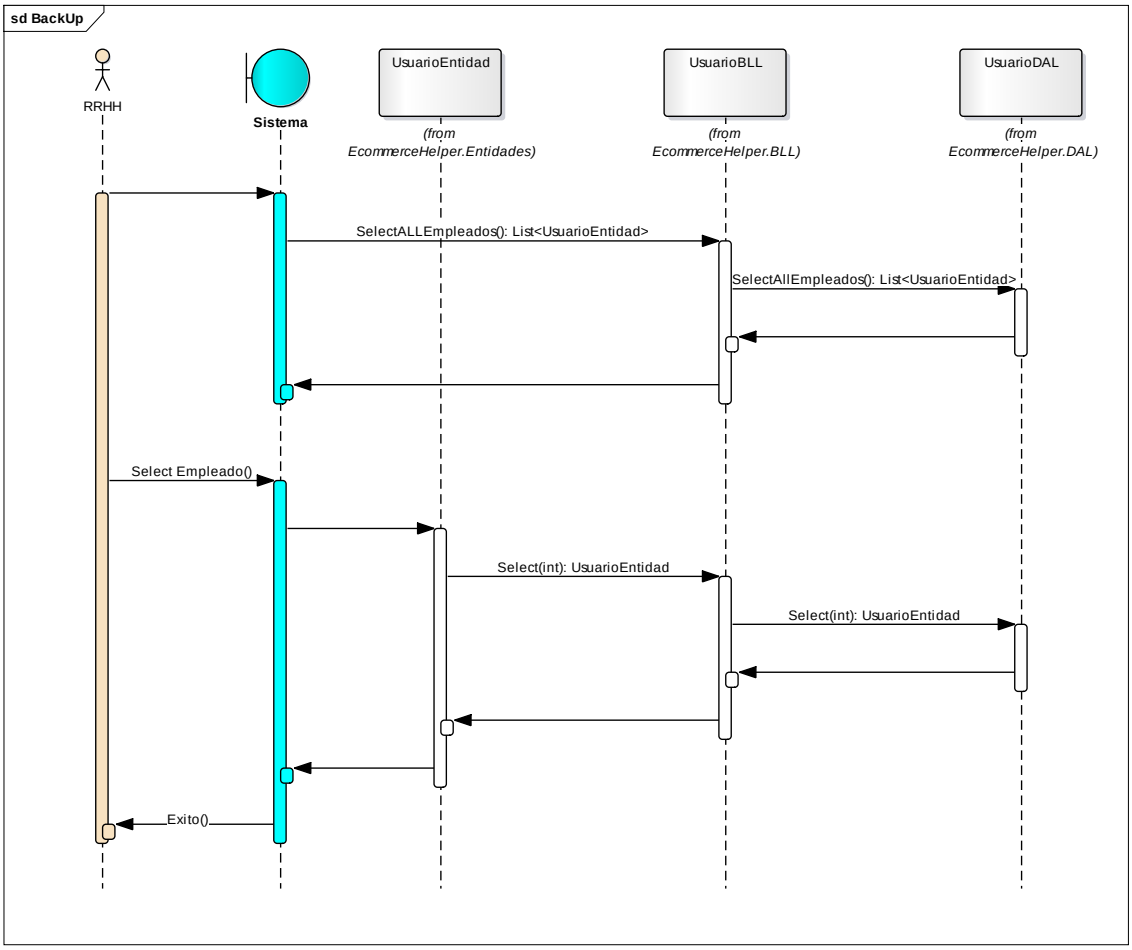

class EcommerceHelper
EcommerceHelper.Entidades::FamiliaEntidad
«property»
+ IdFamilia(): int
+ NombreFamilia(): string
-_Familia
EcommerceHelper.Entidades::PatenteEntidad
«property»
+ Descripcion(): string
+ IdPatente(): int
+_Permisos
EcommerceHelper.Entidades::UsuarioEntidad
- _Familia: FamiliaEntidad = new FamiliaEntidad()
- _Permisos: List<IFamPat> = new List<IFamPat>()
«property»
+ Apellido(): string
+ DVH(): int
+ Email(): string
+ Familia(): FamiliaEntidad
+ IdTipoUsuario(): int
+ IdUsuario(): int
+ MiDireccion(): List<DireccionEntidad >
+ MiDocumento(): TipoDeDocumentoEntidad
+ MiSexo(): SexoEntidad
+ MiTelefono(): TipoTelefonoEntidad
+ MiUsuario(): TipoUsuarioEntidad
+ Nombre(): string
+ NumeroDocumento(): int
+ NumeroTelefono(): Int64
+ Password(): string
+ Permisos(): List<IFamPat>
0..*
1
EcommerceHelper.Entidades::TipoTelefonoEntidad
«property»
+ Descripcion(): string
+ IdTipoTelefono(): int
1
EcommerceHelper.Entidades::SexoEntidad
«property»
+ Descripcion(): string
+ IdSexo(): int
1
1..*
1
EcommerceHelper.Entidades::DireccionEntidad
+ _MiLocalidad: LocalidadEntidad = new LocalidadEn...
+ _MiProvincia: ProvinciaEntidad = new ProvinciaEn...
+ _MiTipoDireccion: TipoDireccionEntidad = new TipoDirecci...
«property»
+ Calle(): string
+ Departamento(): string
+ IdDireccion(): int
+ MiLocalidad(): LocalidadEntidad
+ MiProvincia(): ProvinciaEntidad
+ MiTipoDireccion(): TipoDireccionEntidad
+ Numero(): int
+ Piso(): string
«use»
EcommerceHelper.BLL::UsuarioBLL
- _DalDireccion: DireccionDAL
- _dalProvicia: ProvinciaDAL
- _DalUsuario: UsuarioDAL
+ ActualizarPassword(string, string): UsuarioEntidad
+ BuscarMail(string): UsuarioEntidad
+ EliminarDireccion(int, int): void
+ GenerarCadenaDVHTablaEntera(): List< UsuarioEntidad>
+ IniciarSesion(string, string): UsuarioEntidad
+ InsertDireccionDeFacturacion(DireccionEntidad, UsuarioEntidad): void
+ InsertFamiliaUsuario(int, int, string): int
+ RegistrarUsuario(UsuarioEntidad): int
+ Select(int): UsuarioEntidad
+ SelectALLEmpleadoDeLimpieza(): List<UsuarioEntidad>
+ SelectALLEmpleadoDeLimpiezabyLocalidad(int): List<UsuarioEntidad>
+ SelectALLEmpleados(): List<UsuarioEntidad>
+ SelectALLProvincias(): List<ProvinciaEntidad>
+ SelectALLSexos(): List<SexoEntidad>
+ UpdateDatosEmpleado(UsuarioEntidad): void
+ UpdateDireccionEmpleado(DireccionEntidad, UsuarioEntidad): void
+ UsuarioBLL()
+ UsuarioModificarPermisos(List<IFamPat>, List<IFamPat>, string): bool
+ UsuarioTraerPermisos(string, int): List<IFamPat>
+_MiLocalidad
EcommerceHelper.Entidades::LocalidadEntidad
«property»
+ Descripcion(): string
+ IdLocalidad(): int
+ MiProvincia(): ProvinciaEntidad

Diagrama de Secuencia CU - 009 – Consultar Empleado



Pantalla

Listado de Empleados

Nombre	Apellido	NumeroDocumento	NumeroTelefono	IdSexo	Email	IdDireccion	Calle	Numero	Piso	Departamento	IdLocalidad	IdProvincia		
leandro	messiano	33897289	237465898	2	messi3@gmail.com	16	avellaneda	876	0	0	4	1	Modificar	Eliminar
Susana	Alegria	23878111	237477188	1	alegriasusana@gmail.com	37	escueles	234	0	a	4	1	Modificar	Eliminar
leandro	messiano	33897289	237465898	2	messi3@gmail.com	38	Florencio Sánchez	1300	0	0	2	1	Modificar	Eliminar
Pablo Martin	Strasser	27666555	237484998	2	PabloMartin@gmail.com	40	Int. Luis Tulissi	1930	0	0	3	1	Modificar	Eliminar

Requisitos relacionados organización de Recursos

- CU - 010 - Generar orden de trabajo
- CU - 011 – Asignación de Orden de Trabajo
- CU - 012 – Estadísticas de trabajos
- CU -044 – Modificar Orden de Trabajo

CU - 010 - Generar orden de trabajo

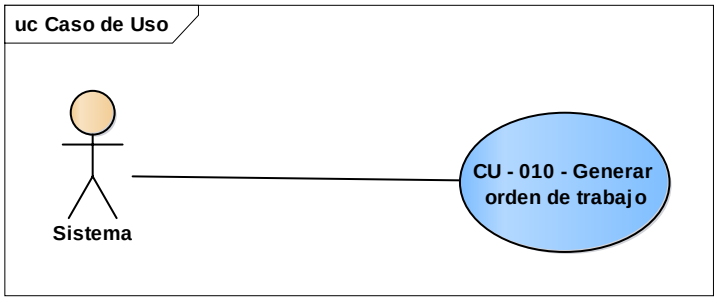
Precondición
El usuario cliente debe estar iniciado sesión y haber seleccionado comprar un servicio.

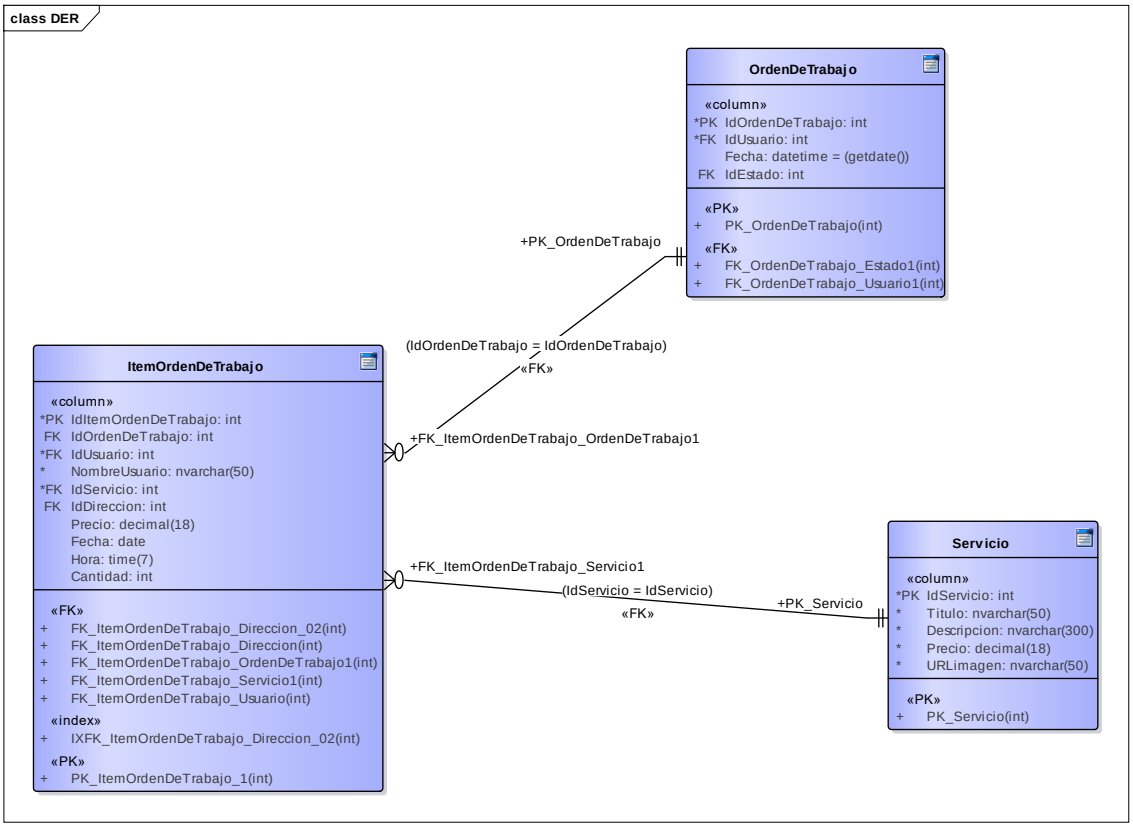
Actores primarios:
Cliente

Camino principal del caso de uso – Generar Orden de Trabajo

1 El sistema genera una orden de trabajo con estado activa.
2 El sistema genera un ítem de orden de trabajo donde se incluye el servicio seleccionado por el cliente.

Postcondición:
Queda confecciona una Orden de trabajo con el ítem correspondiente al servicio.


uc Caso de Uso
Sistema
CU - 010 - Generar orden de trabajo


class DER
OrdenDeTrabajo
«column»
*PK IdOrdenDeTrabajo: int
*FK IdUsuario: int
Fecha: datetime = (getdate())
FK IdEstado: int
«PK»
+ PK_OrdenDeTrabajo(int)
«FK»
+ FK_OrdenDeTrabajo_Estado1(int)
+ FK_OrdenDeTrabajo_Usuario1(int)
+PK_OrdenDeTrabajo
(IdOrdenDeTrabajo = IdOrdenDeTrabajo)
«FK»
ItemOrdenDeTrabajo
«column»
*PK IdItemOrdenDeTrabajo: int
FK IdOrdenDeTrabajo: int
*FK IdUsuario: int
* NombreUsuario: nvarchar(50)
*FK IdServicio: int
FK IdDireccion: int
Precio: decimal(18)
Fecha: date
Hora: time(7)
Cantidad: int
«FK»
+ FK_ItemOrdenDeTrabajo_Direccion_02(int)
+ FK_ItemOrdenDeTrabajo_Direccion(int)
+ FK_ItemOrdenDeTrabajo_OrdenDeTrabajo1(int)
+ FK_ItemOrdenDeTrabajo_Servicio1(int)
+ FK_ItemOrdenDeTrabajo_Usuario(int)
«index»
+ IXFK_ItemOrdenDeTrabajo_Direccion_02(int)
«PK»
+ PK_ItemOrdenDeTrabajo_1(int)
+FK_ItemOrdenDeTrabajo_OrdenDeTrabajo1
+FK_ItemOrdenDeTrabajo_Servicio1
(IdServicio = IdServicio)
«FK»
+PK_Servicio
Servicio
«column»
*PK IdServicio: int
* Titulo: nvarchar(50)
* Descripcion: nvarchar(300)
* Precio: decimal(18)
* URLimagen: nvarchar(50)
«PK»
+ PK_Servicio(int)

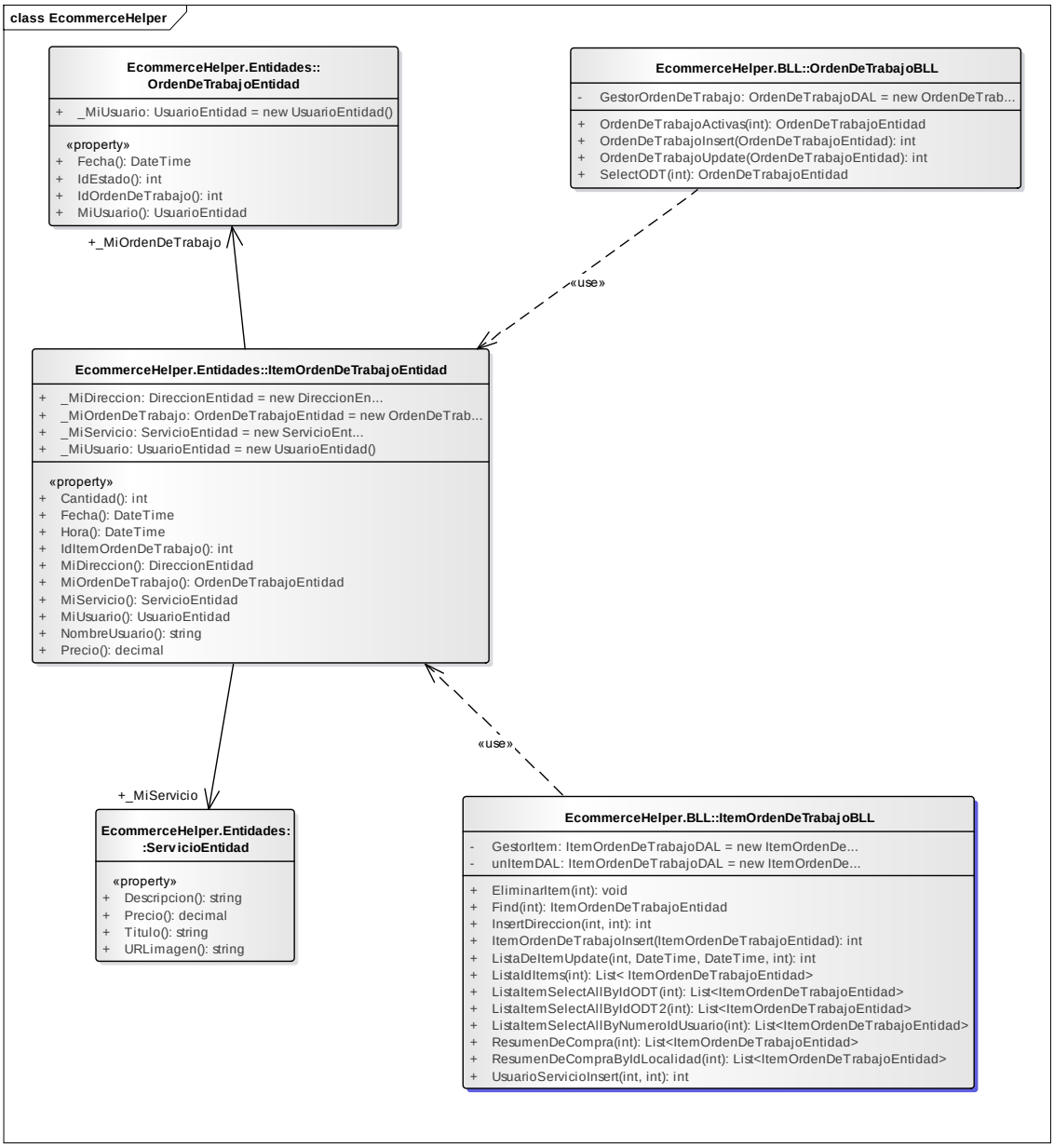
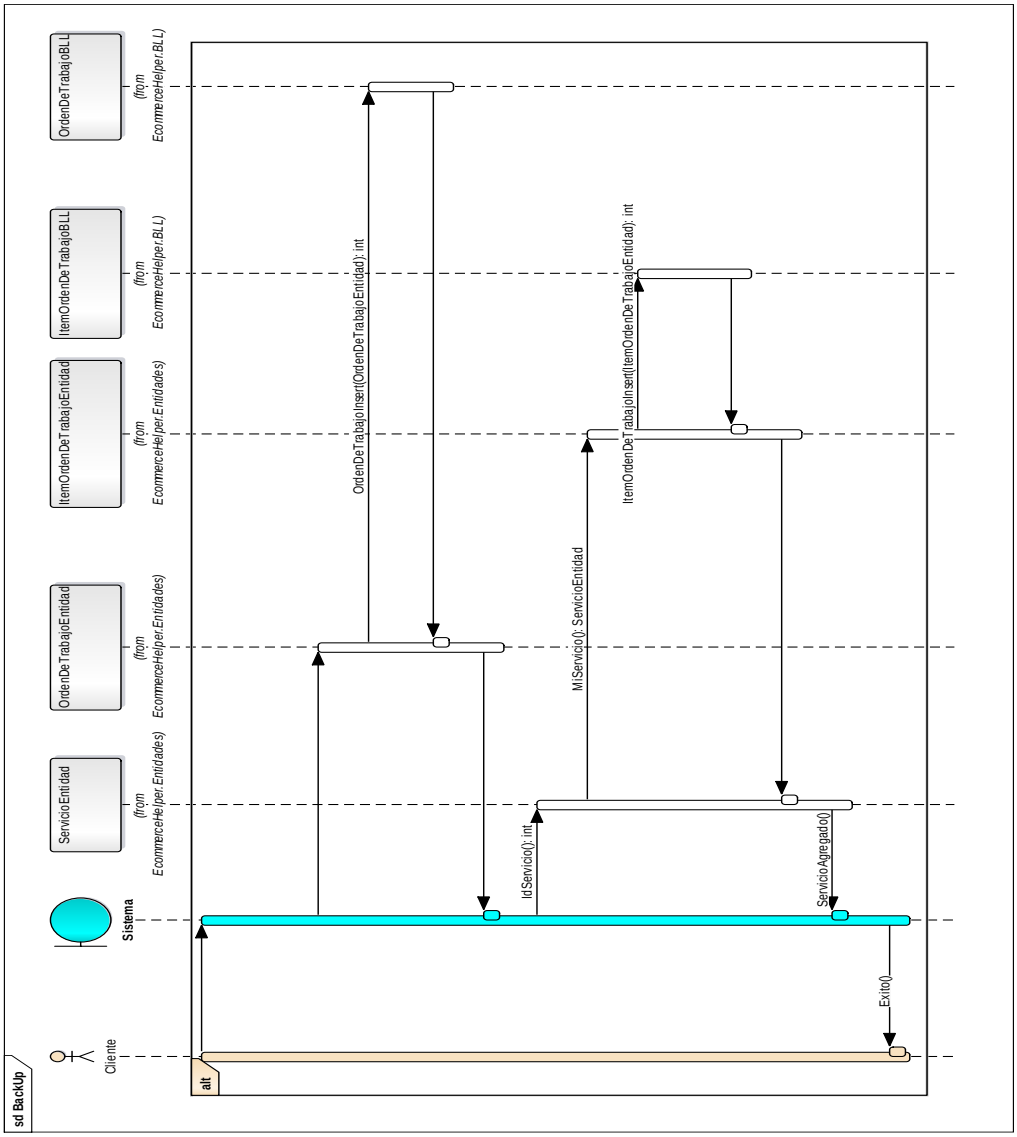

class EcommerceHelper
EcommerceHelper.Entidades::OrdenDeTrabajoEntidad
+ _MiUsuario: UsuarioEntidad = new UsuarioEntidad()
«property»
+ Fecha(): DateTime
+ IdEstado(): int
+ IdOrdenDeTrabajo(): int
+ MiUsuario(): UsuarioEntidad
EcommerceHelper.BLL::OrdenDeTrabajoBLL
- GestorOrdenDeTrabajo: OrdenDeTrabajoDAL = new OrdenDeTrab...
+ OrdenDeTrabajoActivas(int): OrdenDeTrabajoEntidad
+ OrdenDeTrabajoInsert(OrdenDeTrabajoEntidad): int
+ OrdenDeTrabajoUpdate(OrdenDeTrabajoEntidad): int
+ SelectODT(int): OrdenDeTrabajoEntidad
+_MiOrdenDeTrabajo
«use»
EcommerceHelper.Entidades::ItemOrdenDeTrabajoEntidad
+ _MiDireccion: DireccionEntidad = new DireccionEn...
+ _MiOrdenDeTrabajo: OrdenDeTrabajoEntidad = new OrdenDeTrab...
+ _MiServicio: ServicioEntidad = new ServicioEnt...
+ _MiUsuario: UsuarioEntidad = new UsuarioEntidad()
«property»
+ Cantidad(): int
+ Fecha(): DateTime
+ Hora(): DateTime
+ IdItemOrdenDeTrabajo(): int
+ MiDireccion(): DireccionEntidad
+ MiOrdenDeTrabajo(): OrdenDeTrabajoEntidad
+ MiServicio(): ServicioEntidad
+ MiUsuario(): UsuarioEntidad
+ NombreUsuario(): string
+ Precio(): decimal
«use»
+_MiServicio
EcommerceHelper.Entidades::ServicioEntidad
«property»
+ Descripcion(): string
+ Precio(): decimal
+ Titulo(): string
+ URLimagen(): string
EcommerceHelper.BLL::ItemOrdenDeTrabajoBLL
- GestorItem: ItemOrdenDeTrabajoDAL = new ItemOrdenDe...
- unItemDAL: ItemOrdenDeTrabajoDAL = new ItemOrdenDe...
+ EliminarItem(int): void
+ Find(int): ItemOrdenDeTrabajoEntidad
+ InsertDireccion(int, int): int
+ ItemOrdenDeTrabajoInsert(ItemOrdenDeTrabajoEntidad): int
+ ListaDeItemUpdate(int, DateTime, DateTime, int): int
+ ListaIdItems(int): List< ItemOrdenDeTrabajoEntidad>
+ ListaItemSelectAllByIdODT(int): List<ItemOrdenDeTrabajoEntidad>
+ ListaItemSelectAllByIdODT2(int): List<ItemOrdenDeTrabajoEntidad>
+ ListaItemSelectAllByNumeroIdUsuario(int): List<ItemOrdenDeTrabajoEntidad>
+ ResumenDeCompra(int): List<ItemOrdenDeTrabajoEntidad>
+ ResumenDeCompraByIdLocalidad(int): List<ItemOrdenDeTrabajoEntidad>
+ UsuarioServicioInsert(int, int): int

Diagrama de Secuencia CU – 010 – Realizar de Orden de Trabajo



CU - 011 – Asignación de Orden de Trabajo

Precondición
La orden de trabajo debe estar generada

Actores primarios:
Jefe de ventas

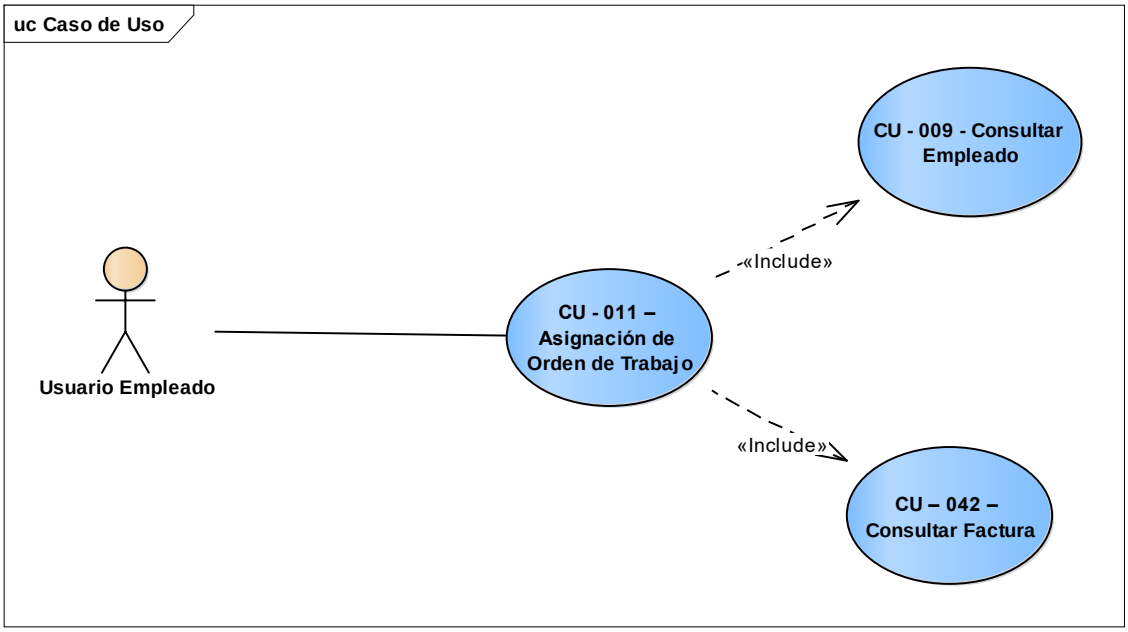
Camino principal del caso de uso – Asignar Orden de Trabajo

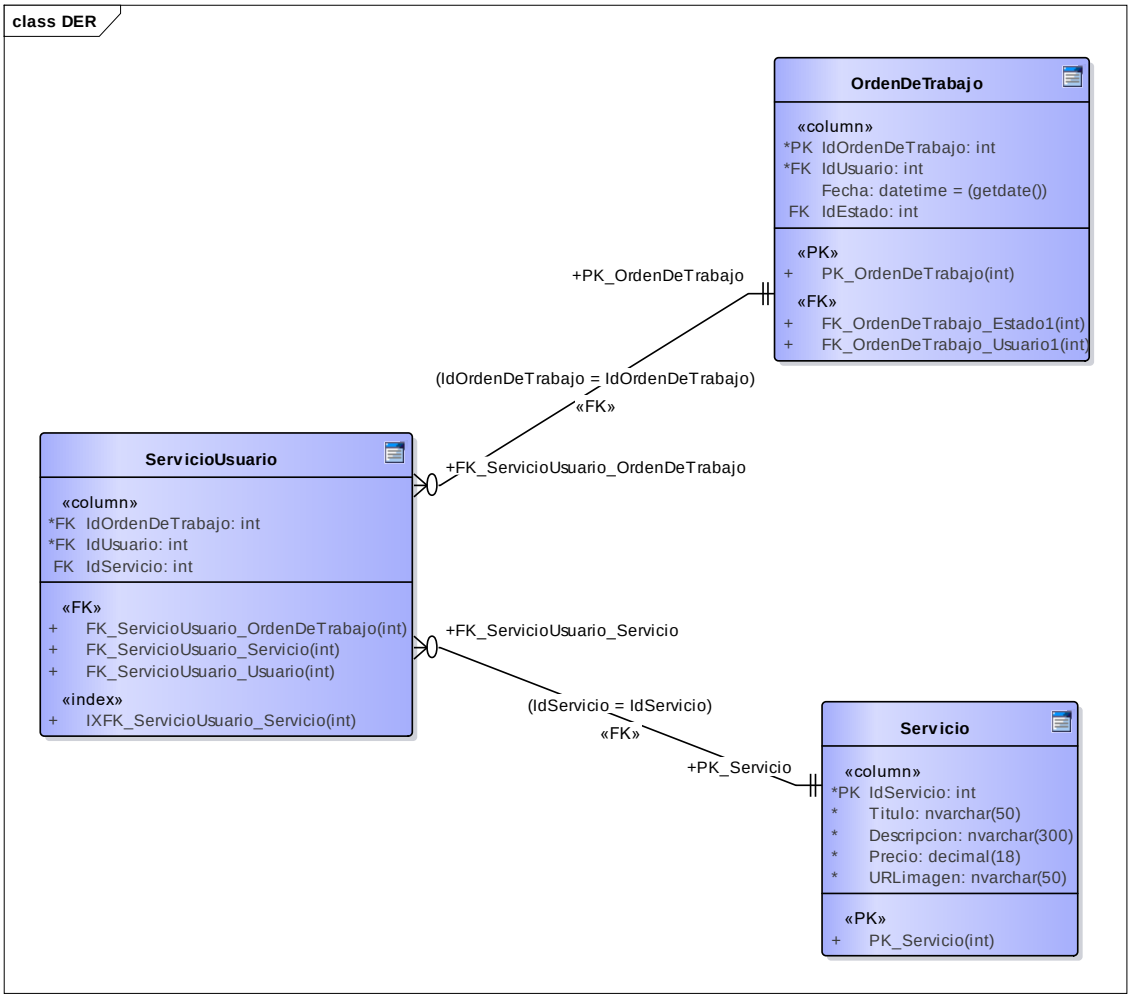
1. El jefe de ventas selecciona asignar orden de trabajo.
2. El sistema solicita que se selecciona la orden de trabajo a asignar.
3. El jefe de ventas visualiza los datos que el sistema le desplego.
4. El sistema despliega la lista de empleados que se encuentran más cerca del domicilio donde se llevara adelante el servicio.
5. El sistema solicita que seleccione el empleado para realizar dicha tarea.
6. El sistema agrega dicha tarea al empleado y a la Orden de trabajo al empleado.
7. El jefe de ventas selecciona finalizar.
8. El sistema refresca la pantalla con las nuevas órdenes de trabajo asignadas.

Caminos alternativos

No existen caminos alternativos.

Postcondición:
Queda asignada la Orden de trabajo.


uc Caso de Uso
CU - 009 - Consultar Empleado
«Include»
Usuario Empleado
CU - 011 – Asignación de Orden de Trabajo
«Include»
CU – 042 – Consultar Factura


class DER
OrdenDeTrabajo
«column»
*PK IdOrdenDeTrabajo: int
*FK IdUsuario: int
Fecha: datetime = (getdate())
FK IdEstado: int
«PK»
+ PK_OrdenDeTrabajo(int)
«FK»
+ FK_OrdenDeTrabajo_Estado1(int)
+ FK_OrdenDeTrabajo_Usuario1(int)
+PK_OrdenDeTrabajo
(IdOrdenDeTrabajo = IdOrdenDeTrabajo)
«FK»
+FK_ServicioUsuario_OrdenDeTrabajo
ServicioUsuario
«column»
*FK IdOrdenDeTrabajo: int
*FK IdUsuario: int
FK IdServicio: int
«FK»
+ FK_ServicioUsuario_OrdenDeTrabajo(int)
+ FK_ServicioUsuario_Servicio(int)
+ FK_ServicioUsuario_Usuario(int)
«index»
+ IXFK_ServicioUsuario_Servicio(int)
+FK_ServicioUsuario_Servicio
(IdServicio = IdServicio)
«FK»
+PK_Servicio
Servicio
«column»
*PK IdServicio: int
* Titulo: nvarchar(50)
* Descripcion: nvarchar(300)
* Precio: decimal(18)
* URLimagen: nvarchar(50)
«PK»
+ PK_Servicio(int)

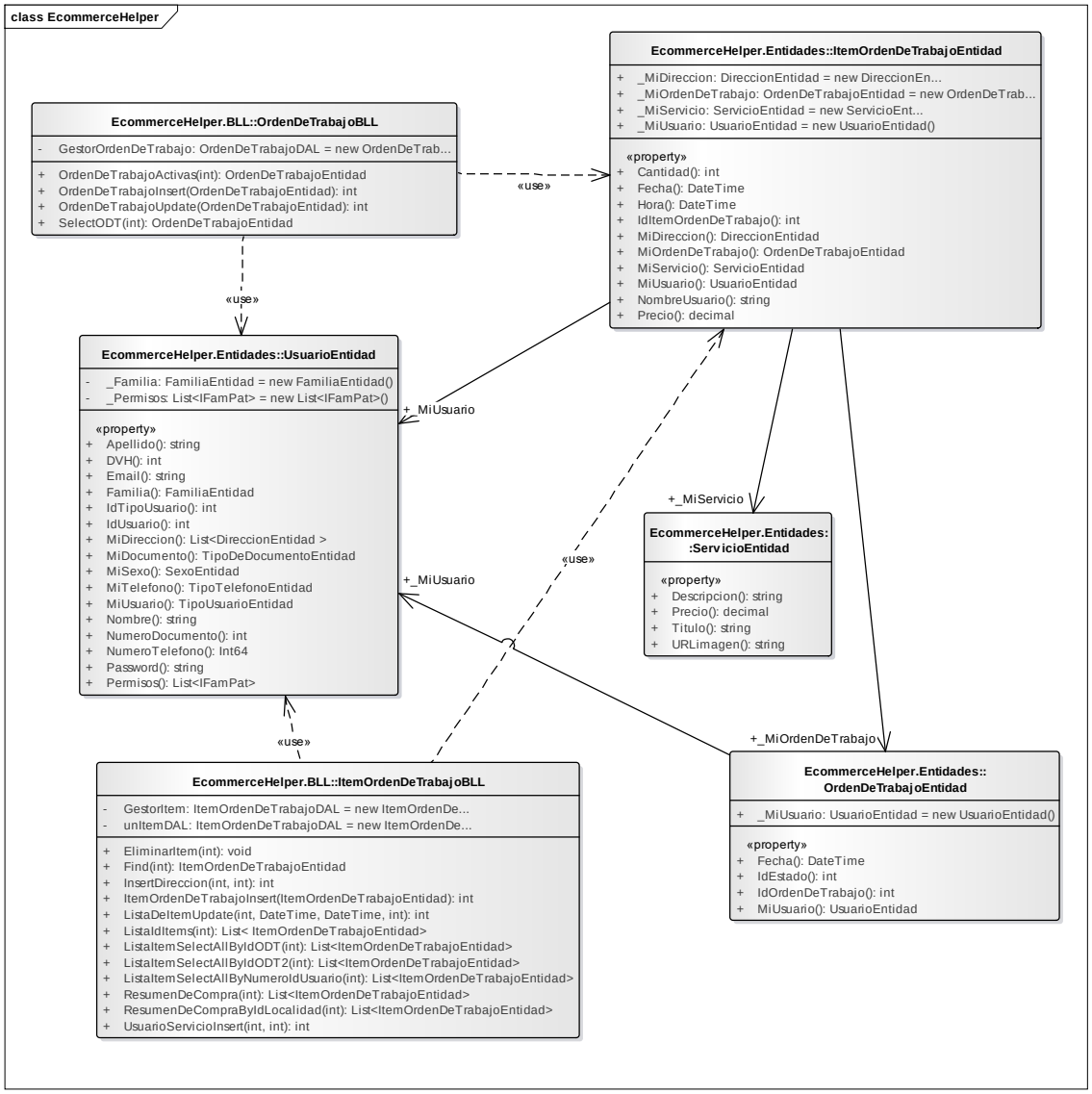
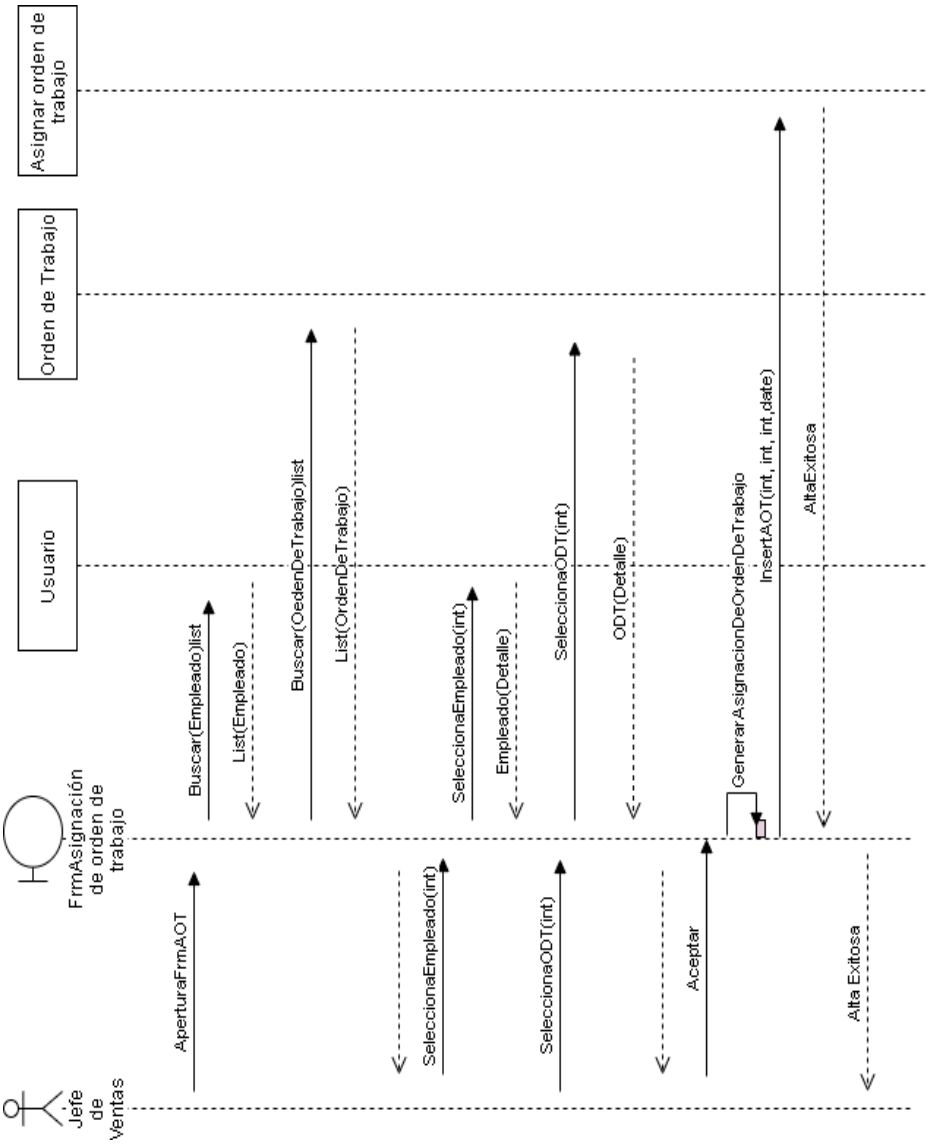
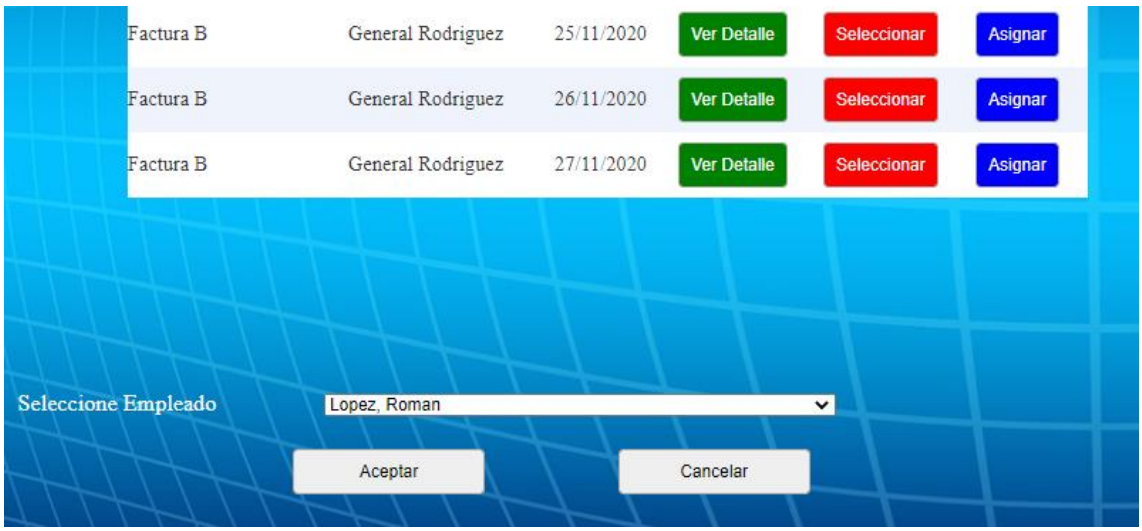

class EcommerceHelper
EcommerceHelper.Entidades::ItemOrdenDeTrabajoEntidad
+ _MiDireccion: DireccionEntidad = new DireccionEn...
+ _MiOrdenDeTrabajo: OrdenDeTrabajoEntidad = new OrdenDeTrab...
+ _MiServicio: ServicioEntidad = new ServicioEnt...
+ _MiUsuario: UsuarioEntidad = new UsuarioEntidad()
«property»
+ Cantidad(): int
+ Fecha(): DateTime
+ Hora(): DateTime
+ IdItemOrdenDeTrabajo(): int
+ MiDireccion(): DireccionEntidad
+ MiOrdenDeTrabajo(): OrdenDeTrabajoEntidad
+ MiServicio(): ServicioEntidad
+ MiUsuario(): UsuarioEntidad
+ NombreUsuario(): string
+ Precio(): decimal
EcommerceHelper.BLL::OrdenDeTrabajoBLL
- GestorOrdenDeTrabajo: OrdenDeTrabajoDAL = new OrdenDeTrab...
+ OrdenDeTrabajoActivas(int): OrdenDeTrabajoEntidad
+ OrdenDeTrabajoInsert(OrdenDeTrabajoEntidad): int
+ OrdenDeTrabajoUpdate(OrdenDeTrabajoEntidad): int
+ SelectODT(int): OrdenDeTrabajoEntidad
«use»
«use»
EcommerceHelper.Entidades::UsuarioEntidad
- _Familia: FamiliaEntidad = new FamiliaEntidad()
- _Permisos: List<IFamPat> = new List<IFamPat>()
«property»
+ Apellido(): string
+ DVH(): int
+ Email(): string
+ Familia(): FamiliaEntidad
+ IdTipoUsuario(): int
+ IdUsuario(): int
+ MiDireccion(): List<DireccionEntidad >
+ MiDocumento(): TipoDeDocumentoEntidad
+ MiSexo(): SexoEntidad
+ MiTelefono(): TipoTelefonoEntidad
+ MiUsuario(): TipoUsuarioEntidad
+ Nombre(): string
+ NumeroDocumento(): int
+ NumeroTelefono(): Int64
+ Password(): string
+ Permisos(): List<IFamPat>
+_MiUsuario
+_MiUsuario
+_MiServicio
EcommerceHelper.Entidades::ServicioEntidad
«property»
+ Descripcion(): string
+ Precio(): decimal
+ Titulo(): string
+ URLimagen(): string
«use»
«use»
+_MiOrdenDeTrabajo
EcommerceHelper.BLL::ItemOrdenDeTrabajoBLL
- GestorItem: ItemOrdenDeTrabajoDAL = new ItemOrdenDe...
- unItemDAL: ItemOrdenDeTrabajoDAL = new ItemOrdenDe...
+ EliminarItem(int): void
+ Find(int): ItemOrdenDeTrabajoEntidad
+ InsertDireccion(int, int): int
+ ItemOrdenDeTrabajoInsert(ItemOrdenDeTrabajoEntidad): int
+ ListaDeItemUpdate(int, DateTime, DateTime, int): int
+ ListaIdItems(int): List< ItemOrdenDeTrabajoEntidad>
+ ListaItemSelectAllByIdODT(int): List<ItemOrdenDeTrabajoEntidad>
+ ListaItemSelectAllByIdODT2(int): List<ItemOrdenDeTrabajoEntidad>
+ ListaItemSelectAllByNumeroIdUsuario(int): List<ItemOrdenDeTrabajoEntidad>
+ ResumenDeCompra(int): List<ItemOrdenDeTrabajoEntidad>
+ ResumenDeCompraByIdLocalidad(int): List<ItemOrdenDeTrabajoEntidad>
+ UsuarioServicioInsert(int, int): int
EcommerceHelper.Entidades::OrdenDeTrabajoEntidad
+ _MiUsuario: UsuarioEntidad = new UsuarioEntidad()
«property»
+ Fecha(): DateTime
+ IdEstado(): int
+ IdOrdenDeTrabajo(): int
+ MiUsuario(): UsuarioEntidad

Diagrama de Secuencia CU – 011 – Asignar de Orden de Trabajo



Pantalla


Factura B
General Rodriguez
25/11/2020
Ver Detalle
Seleccionar
Asignar
Factura B
General Rodriguez
26/11/2020
Ver Detalle
Seleccionar
Asignar
Factura B
General Rodriguez
27/11/2020
Ver Detalle
Seleccionar
Asignar
Seleccione Empleado
Lopez, Roman
Aceptar
Cancelar

CU - 012 – Estadísticas de trabajos

Precondición
Usuario con sesión iniciada de acuerdo con el perfil.

Actores primarios:
Jefe de ventas / administrativo.

Camino principal del caso de uso – Estadísticas de Trabajo

1) El Usuario selecciona la opción del menú, Historial de Ventas.
2) El Sistema carga la página correspondiente a la generación de reporte de trabajos realizados.
3) El Usuario ingresa selecciona el tipo de consulta para la confección del reporte de ventas:
   a - Por empleado.
   b – Por fecha.
   C – Por Servicio.

   El usuario empleado selecciona finalizar.
4) El sistema refresca la pantalla con la descripción de historial de acuerdo con los parámetros ingresados por el usuario.

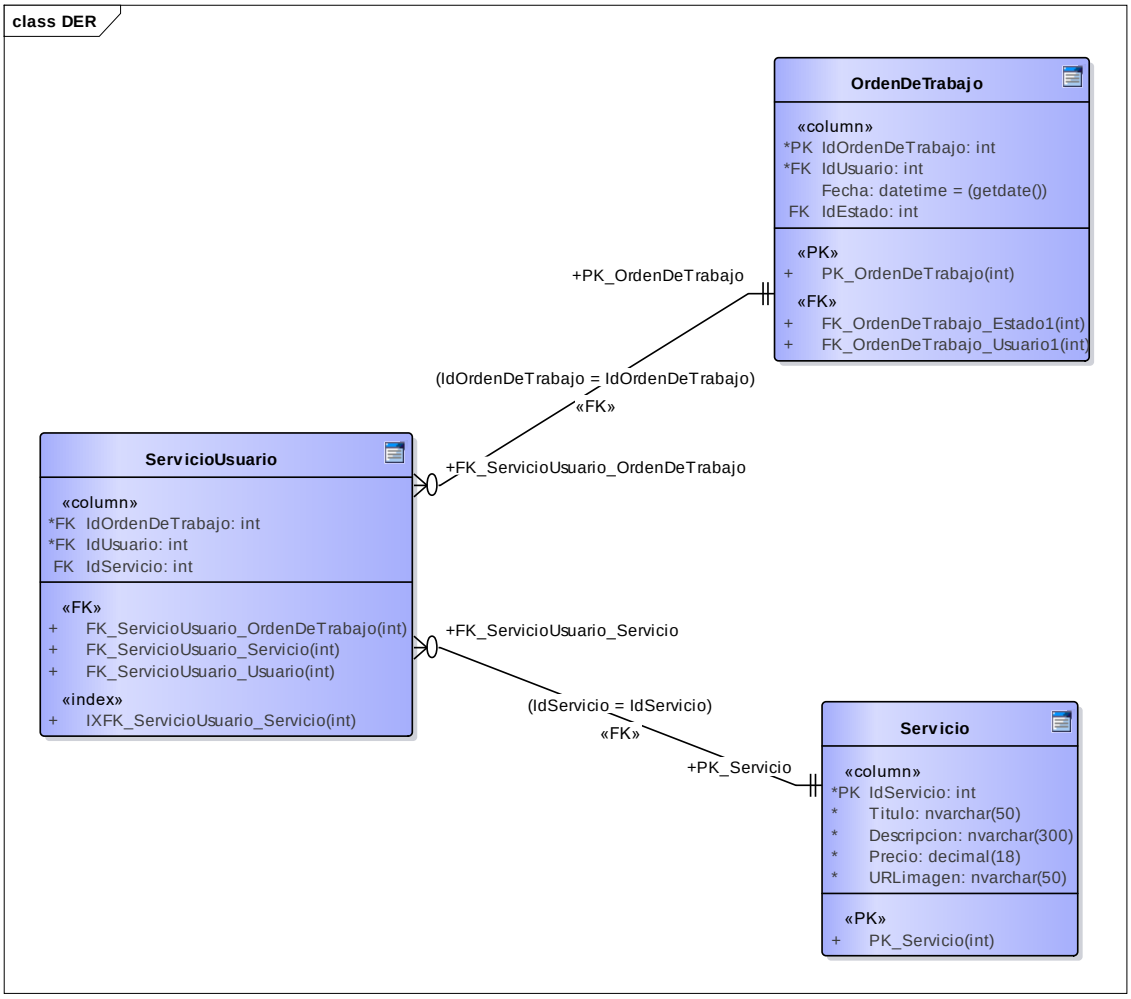
Caminos alternativos

Para el paso 3: El sistema detecta que el empleado seleccionado no posee servicios realizados en su haber.

3.1 El empleado seleccionado no tiene tareas realizadas.
3.2 El sistema informa que el empleado seleccionado no posee tareas realizadas.

Postcondición:
Se visualiza en pantalla el Reporte de Venta generado.

uc Caso de Uso
Usuario Empleado
CU - 012 – Estadísticas de trabajos


class DER
OrdenDeTrabajo
«column»
*PK IdOrdenDeTrabajo: int
*FK IdUsuario: int
Fecha: datetime = (getdate())
FK IdEstado: int
«PK»
+ PK_OrdenDeTrabajo(int)
«FK»
+ FK_OrdenDeTrabajo_Estado1(int)
+ FK_OrdenDeTrabajo_Usuario1(int)
+PK_OrdenDeTrabajo
(IdOrdenDeTrabajo = IdOrdenDeTrabajo)
«FK»
+FK_ServicioUsuario_OrdenDeTrabajo
ServicioUsuario
«column»
*FK IdOrdenDeTrabajo: int
*FK IdUsuario: int
FK IdServicio: int
«FK»
+ FK_ServicioUsuario_OrdenDeTrabajo(int)
+ FK_ServicioUsuario_Servicio(int)
+ FK_ServicioUsuario_Usuario(int)
«index»
+ IXFK_ServicioUsuario_Servicio(int)
+FK_ServicioUsuario_Servicio
(IdServicio = IdServicio)
«FK»
+PK_Servicio
Servicio
«column»
*PK IdServicio: int
* Titulo: nvarchar(50)
* Descripcion: nvarchar(300)
* Precio: decimal(18)
* URLimagen: nvarchar(50)
«PK»
+ PK_Servicio(int)

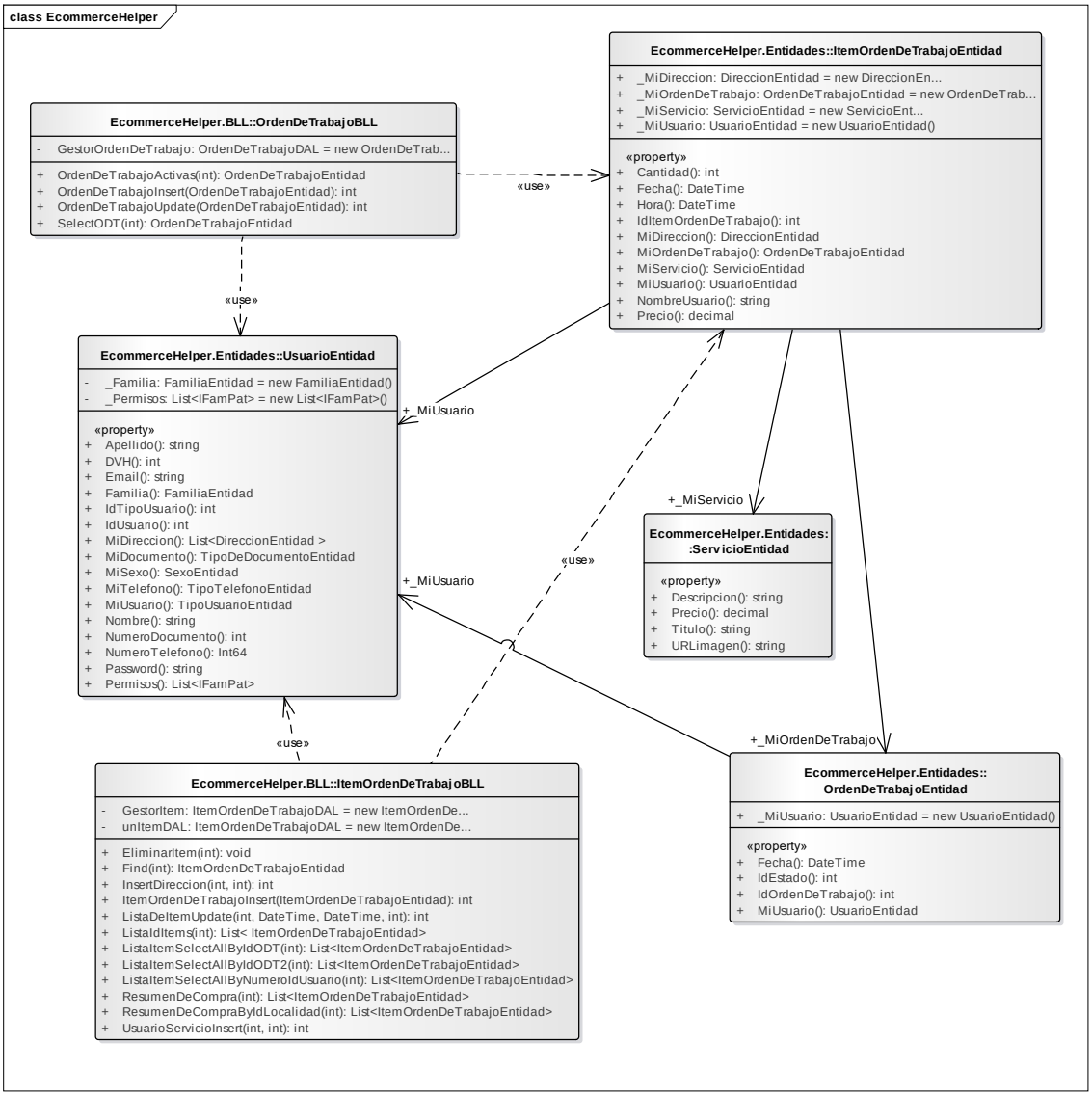
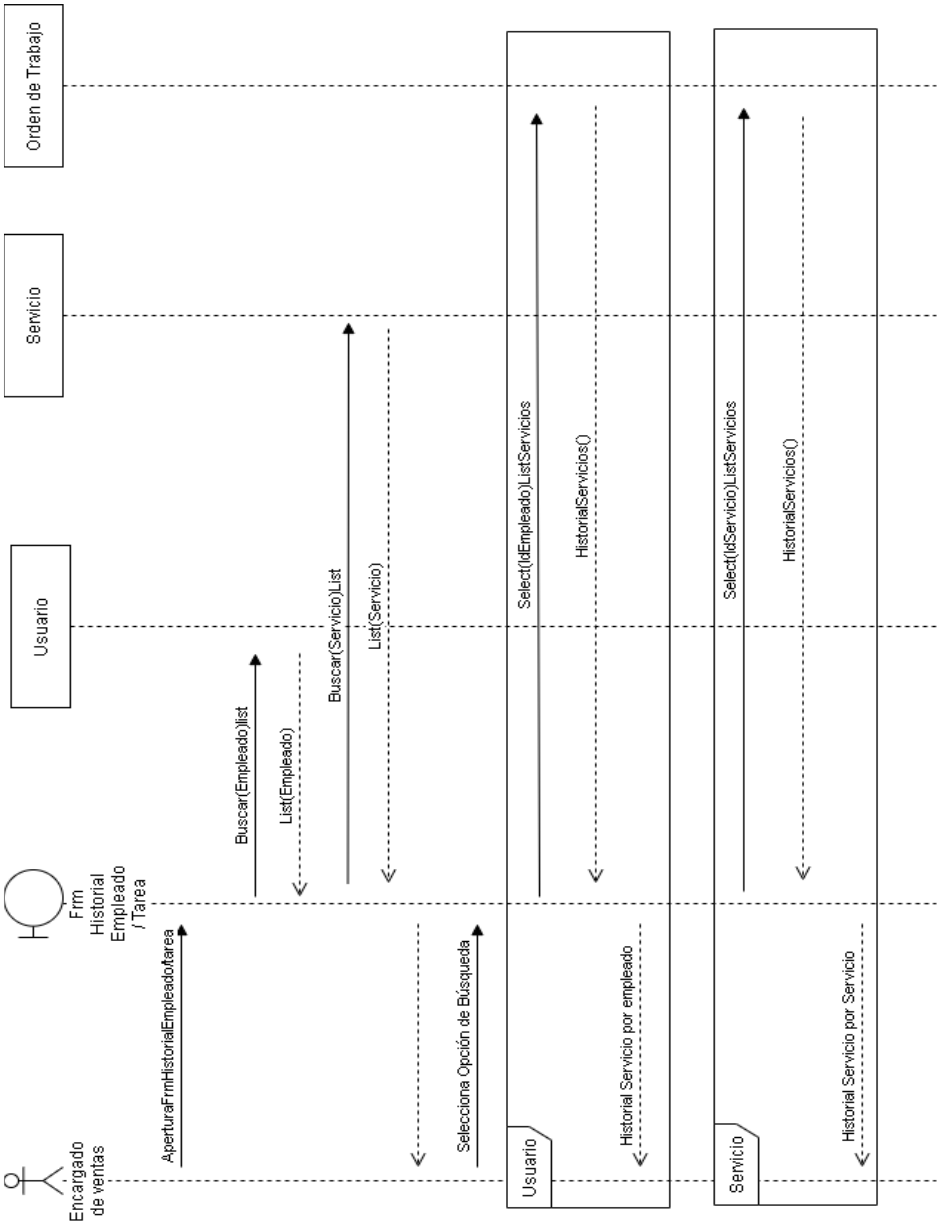

class EcommerceHelper
EcommerceHelper.BLL::OrdenDeTrabajoBLL
- GestorOrdenDeTrabajo: OrdenDeTrabajoDAL = new OrdenDeTrab...
+ OrdenDeTrabajoActivas(int): OrdenDeTrabajoEntidad
+ OrdenDeTrabajoInsert(OrdenDeTrabajoEntidad): int
+ OrdenDeTrabajoUpdate(OrdenDeTrabajoEntidad): int
+ SelectODT(int): OrdenDeTrabajoEntidad
«use»
EcommerceHelper.Entidades::ItemOrdenDeTrabajoEntidad
+ _MiDireccion: DireccionEntidad = new DireccionEn...
+ _MiOrdenDeTrabajo: OrdenDeTrabajoEntidad = new OrdenDeTrab...
+ _MiServicio: ServicioEntidad = new ServicioEnt...
+ _MiUsuario: UsuarioEntidad = new UsuarioEntidad()
«property»
+ Cantidad(): int
+ Fecha(): DateTime
+ Hora(): DateTime
+ IdItemOrdenDeTrabajo(): int
+ MiDireccion(): DireccionEntidad
+ MiOrdenDeTrabajo(): OrdenDeTrabajoEntidad
+ MiServicio(): ServicioEntidad
+ MiUsuario(): UsuarioEntidad
+ NombreUsuario(): string
+ Precio(): decimal
«use»
EcommerceHelper.Entidades::UsuarioEntidad
- _Familia: FamiliaEntidad = new FamiliaEntidad()
- _Permisos: List<IFamPat> = new List<IFamPat>()
«property»
+ Apellido(): string
+ DVH(): int
+ Email(): string
+ Familia(): FamiliaEntidad
+ IdTipoUsuario(): int
+ IdUsuario(): int
+ MiDireccion(): List<DireccionEntidad >
+ MiDocumento(): TipoDeDocumentoEntidad
+ MiSexo(): SexoEntidad
+ MiTelefono(): TipoTelefonoEntidad
+ MiUsuario(): TipoUsuarioEntidad
+ Nombre(): string
+ NumeroDocumento(): int
+ NumeroTelefono(): Int64
+ Password(): string
+ Permisos(): List<IFamPat>
+_MiUsuario
+_MiUsuario
+_MiServicio
EcommerceHelper.Entidades::ServicioEntidad
«property»
+ Descripcion(): string
+ Precio(): decimal
+ Titulo(): string
+ URLimagen(): string
«use»
«use»
+_MiOrdenDeTrabajo
EcommerceHelper.BLL::ItemOrdenDeTrabajoBLL
- GestorItem: ItemOrdenDeTrabajoDAL = new ItemOrdenDe...
- unItemDAL: ItemOrdenDeTrabajoDAL = new ItemOrdenDe...
+ EliminarItem(int): void
+ Find(int): ItemOrdenDeTrabajoEntidad
+ InsertDireccion(int, int): int
+ ItemOrdenDeTrabajoInsert(ItemOrdenDeTrabajoEntidad): int
+ ListaDeItemUpdate(int, DateTime, DateTime, int): int
+ ListaIdItems(int): List< ItemOrdenDeTrabajoEntidad>
+ ListaItemSelectAllByIdODT(int): List<ItemOrdenDeTrabajoEntidad>
+ ListaItemSelectAllByIdODT2(int): List<ItemOrdenDeTrabajoEntidad>
+ ListaItemSelectAllByNumeroIdUsuario(int): List<ItemOrdenDeTrabajoEntidad>
+ ResumenDeCompra(int): List<ItemOrdenDeTrabajoEntidad>
+ ResumenDeCompraByIdLocalidad(int): List<ItemOrdenDeTrabajoEntidad>
+ UsuarioServicioInsert(int, int): int
EcommerceHelper.Entidades::OrdenDeTrabajoEntidad
+ _MiUsuario: UsuarioEntidad = new UsuarioEntidad()
«property»
+ Fecha(): DateTime
+ IdEstado(): int
+ IdOrdenDeTrabajo(): int
+ MiUsuario(): UsuarioEntidad

Diagrama de Secuencia CU - 012 – Estadísticas de trabajos



CU - 044 - Modificar orden de trabajo

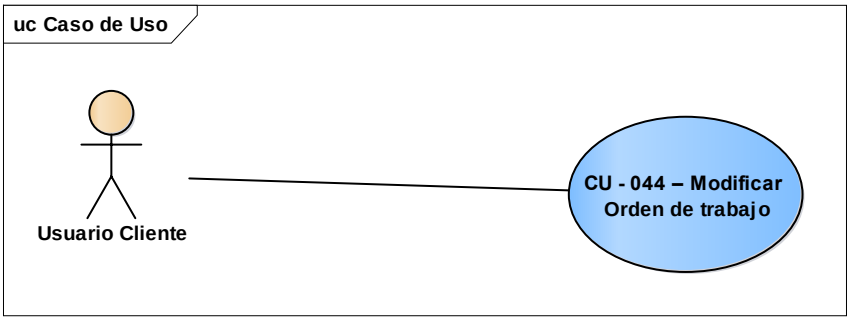
Precondición
El usuario cliente debe estar iniciado sesión y haber comprado al menos un servicio.

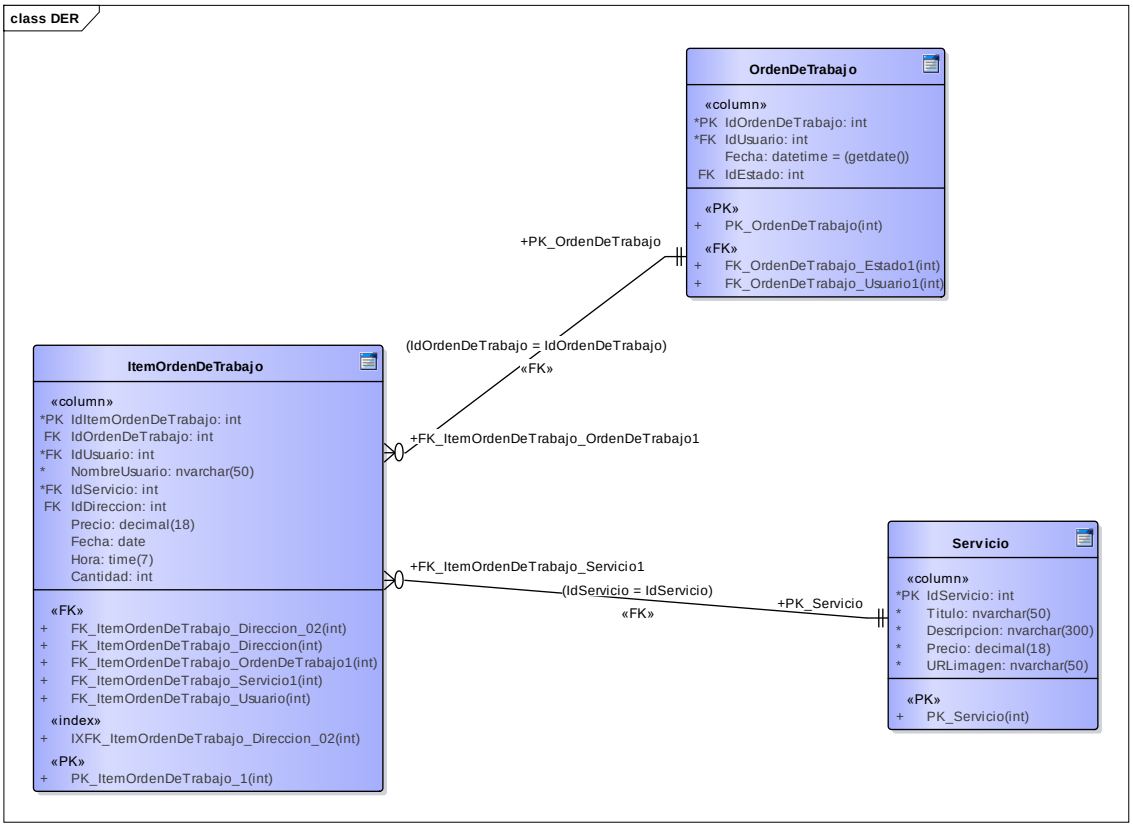
Actores primarios:
Cliente

Camino principal del caso de uso – Modificar Orden de Trabajo

1 El sistema consulta una orden de trabajo con estado activa.
2 El sistema listo un/unos ítem/s de orden de trabajo donde se incluye el servicio seleccionado por el cliente.
3 El cliente selecciona que servicio eliminar.
4 El sistema elimina el ítem.
5 El sistema actualiza la lista de pedido de acuerdo con la orden de trabajo actualizada.

Postcondición:
Queda modifica una Orden de trabajo con el ítem correspondiente al servicio.


uc Caso de Uso
Usuario Cliente
CU - 044 – Modificar Orden de trabajo


class DER
OrdenDeTrabajo
«column»
*PK IdOrdenDeTrabajo: int
*FK IdUsuario: int
Fecha: datetime = (getdate())
FK IdEstado: int
«PK»
+ PK_OrdenDeTrabajo(int)
«FK»
+ FK_OrdenDeTrabajo_Estado1(int)
+ FK_OrdenDeTrabajo_Usuario1(int)
+PK_OrdenDeTrabajo
(IdOrdenDeTrabajo = IdOrdenDeTrabajo)
«FK»
ItemOrdenDeTrabajo
«column»
*PK IdItemOrdenDeTrabajo: int
FK IdOrdenDeTrabajo: int
*FK IdUsuario: int
* NombreUsuario: nvarchar(50)
*FK IdServicio: int
FK IdDireccion: int
Precio: decimal(18)
Fecha: date
Hora: time(7)
Cantidad: int
«FK»
+ FK_ItemOrdenDeTrabajo_Direccion_02(int)
+ FK_ItemOrdenDeTrabajo_Direccion(int)
+ FK_ItemOrdenDeTrabajo_OrdenDeTrabajo1(int)
+ FK_ItemOrdenDeTrabajo_Servicio1(int)
+ FK_ItemOrdenDeTrabajo_Usuario(int)
«index»
+ IXFK_ItemOrdenDeTrabajo_Direccion_02(int)
«PK»
+ PK_ItemOrdenDeTrabajo_1(int)
+FK_ItemOrdenDeTrabajo_OrdenDeTrabajo1
+FK_ItemOrdenDeTrabajo_Servicio1
(IdServicio = IdServicio)
«FK»
+PK_Servicio
Servicio
«column»
*PK IdServicio: int
* Titulo: nvarchar(50)
* Descripcion: nvarchar(300)
* Precio: decimal(18)
* URLimagen: nvarchar(50)
«PK»
+ PK_Servicio(int)

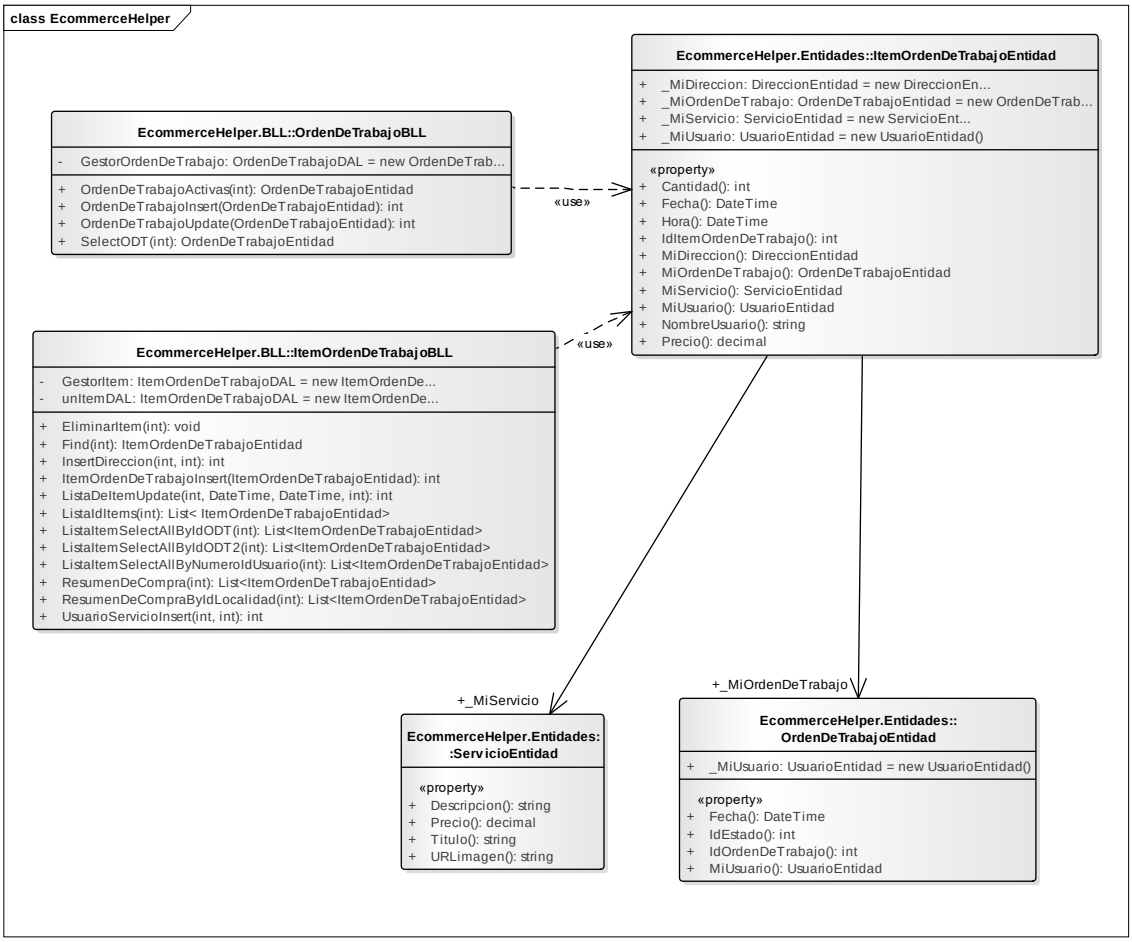
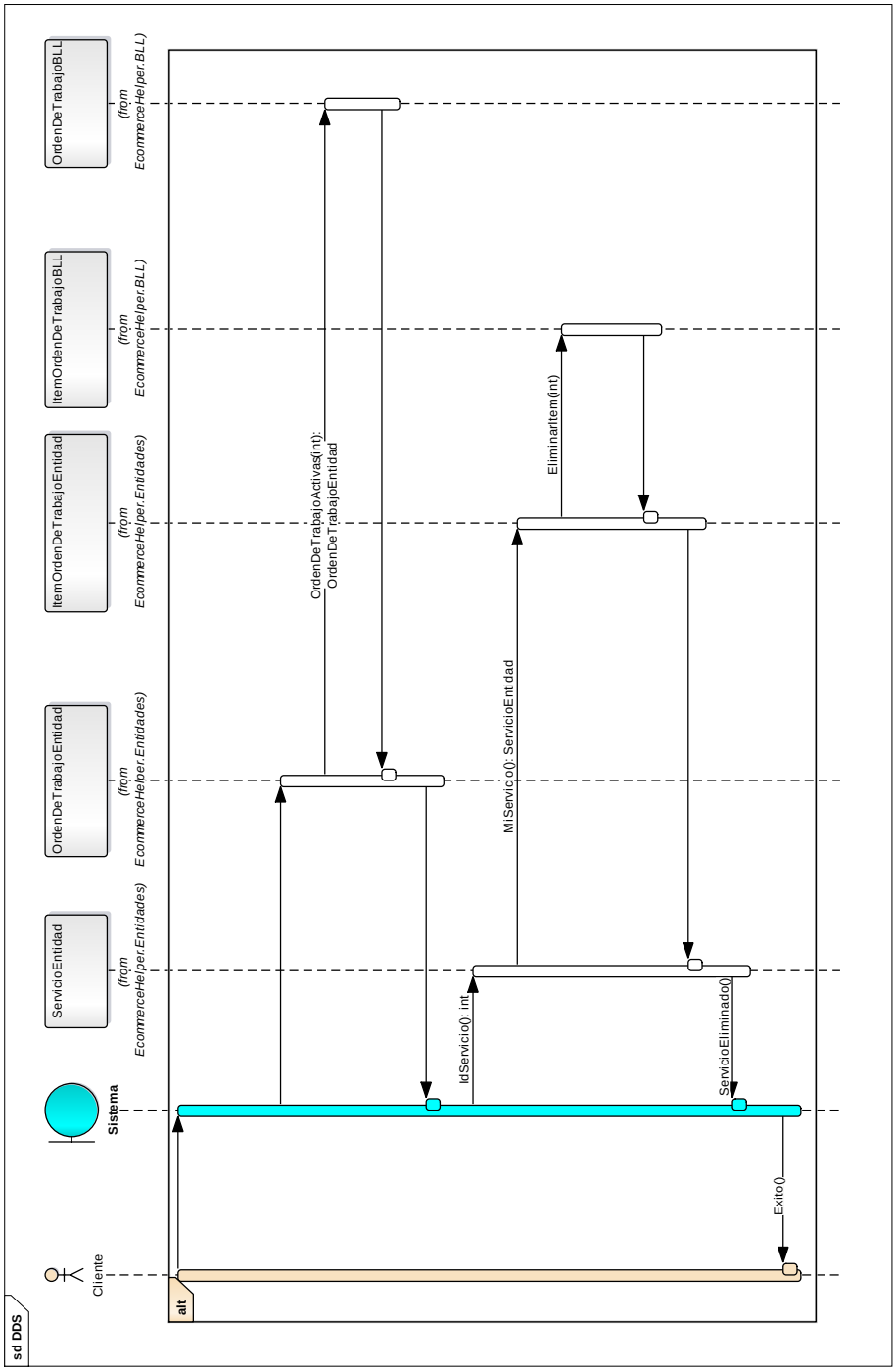

class EcommerceHelper
EcommerceHelper.BLL::OrdenDeTrabajoBLL
- GestorOrdenDeTrabajo: OrdenDeTrabajoDAL = new OrdenDeTrab...
+ OrdenDeTrabajoActivas(int): OrdenDeTrabajoEntidad
+ OrdenDeTrabajoInsert(OrdenDeTrabajoEntidad): int
+ OrdenDeTrabajoUpdate(OrdenDeTrabajoEntidad): int
+ SelectODT(int): OrdenDeTrabajoEntidad
«use»
EcommerceHelper.Entidades::ItemOrdenDeTrabajoEntidad
+ _MiDireccion: DireccionEntidad = new DireccionEn...
+ _MiOrdenDeTrabajo: OrdenDeTrabajoEntidad = new OrdenDeTrab...
+ _MiServicio: ServicioEntidad = new ServicioEnt...
+ _MiUsuario: UsuarioEntidad = new UsuarioEntidad()
«property»
+ Cantidad(): int
+ Fecha(): DateTime
+ Hora(): DateTime
+ IdItemOrdenDeTrabajo(): int
+ MiDireccion(): DireccionEntidad
+ MiOrdenDeTrabajo(): OrdenDeTrabajoEntidad
+ MiServicio(): ServicioEntidad
+ MiUsuario(): UsuarioEntidad
+ NombreUsuario(): string
+ Precio(): decimal
«use»
EcommerceHelper.BLL::ItemOrdenDeTrabajoBLL
- GestorItem: ItemOrdenDeTrabajoDAL = new ItemOrdenDe...
- unItemDAL: ItemOrdenDeTrabajoDAL = new ItemOrdenDe...
+ EliminarItem(int): void
+ Find(int): ItemOrdenDeTrabajoEntidad
+ InsertDireccion(int, int): int
+ ItemOrdenDeTrabajoInsert(ItemOrdenDeTrabajoEntidad): int
+ ListaDeItemUpdate(int, DateTime, DateTime, int): int
+ ListaIdItems(int): List< ItemOrdenDeTrabajoEntidad>
+ ListaItemSelectAllByIdODT(int): List<ItemOrdenDeTrabajoEntidad>
+ ListaItemSelectAllByIdODT2(int): List<ItemOrdenDeTrabajoEntidad>
+ ListaItemSelectAllByNumeroIdUsuario(int): List<ItemOrdenDeTrabajoEntidad>
+ ResumenDeCompra(int): List<ItemOrdenDeTrabajoEntidad>
+ ResumenDeCompraByIdLocalidad(int): List<ItemOrdenDeTrabajoEntidad>
+ UsuarioServicioInsert(int, int): int
+_MiServicio
EcommerceHelper.Entidades::ServicioEntidad
«property»
+ Descripcion(): string
+ Precio(): decimal
+ Titulo(): string
+ URLimagen(): string
+_MiOrdenDeTrabajo
EcommerceHelper.Entidades::OrdenDeTrabajoEntidad
+ _MiUsuario: UsuarioEntidad = new UsuarioEntidad()
«property»
+ Fecha(): DateTime
+ IdEstado(): int
+ IdOrdenDeTrabajo(): int
+ MiUsuario(): UsuarioEntidad

Diagrama de Secuencia CU - 044 – Modificar Orden de trabajo



Requisitos relacionados a Ventas

- CU - 013 - Facturación
- CU - 014 – Emisión de Notas de Crédito
- CU - 015 – Emisión de Notas de Debito
- CU - 017 – Cobranza
- CU – O44 – Generar Recibo
- CU – 041 – Consultar Factura

CU - 013 – Facturación

Precondición
Debe haber una orden de trabajo.

Actores primarios:
Cliente

Camino principal del caso de uso – Facturación

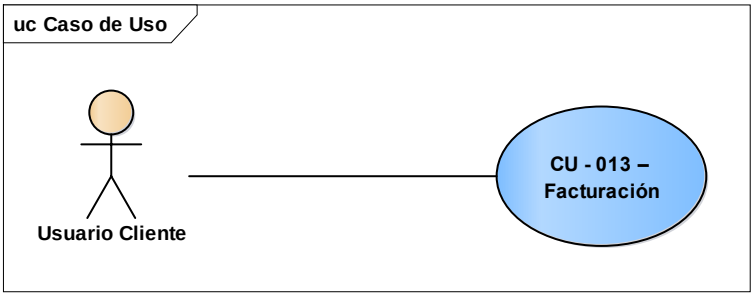
1. El Cliente visualiza el resumen de compra.
2. El sistema muestra la orden de pedido del servicio seleccionada con todos los datos, sus montos y el monto total final en pesos y la dirección donde se llevará a cabo el servicio.
3. El sistema muestra en pantalla los datos que se incluirán en la factura y espera la confirmación para su generación
   Los datos que se muestran son los servicios y cantidad de estos a facturar, precios unitarios y monto total.
4. El cliente selecciona “pagar” por lo tanto confirma el resumen de compra y se genera la factura.
5. El sistema genera la factura del cliente.

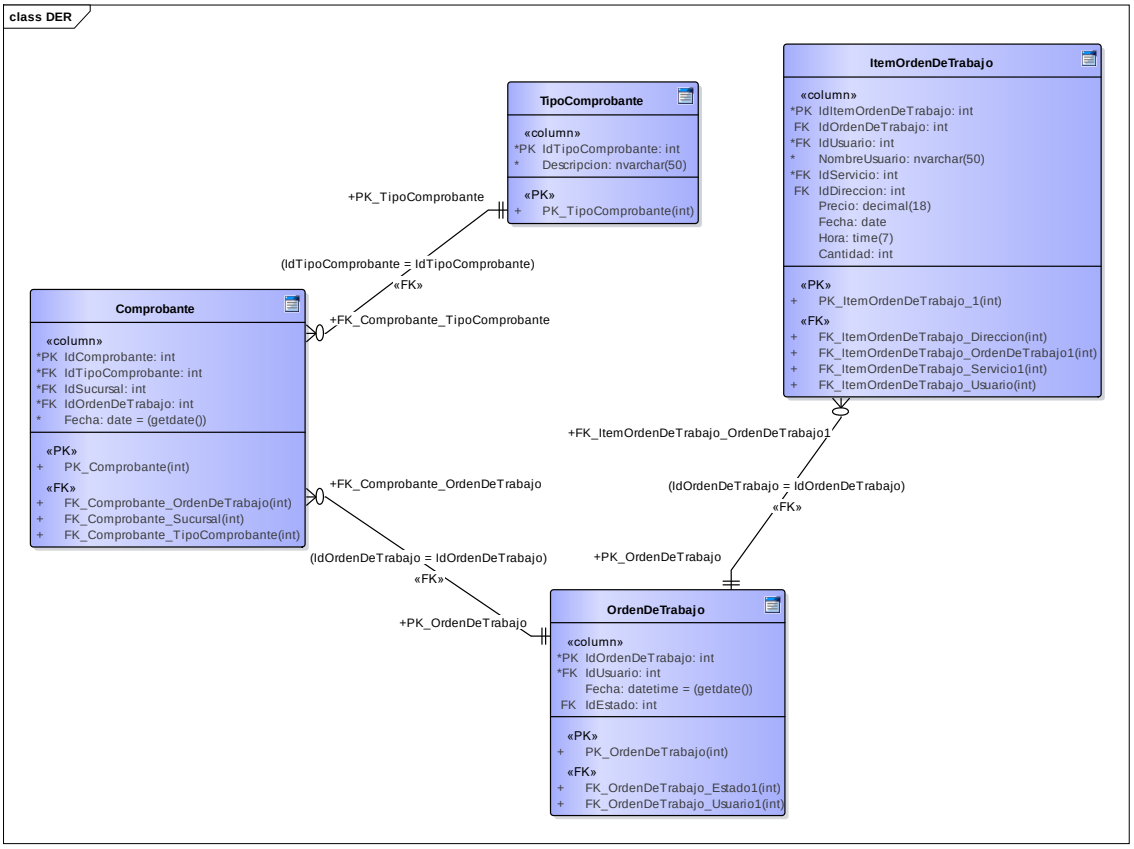
Caminos Alternativos

Para el paso 4: El cliente no se encuentra en el sistema

4 1. El sistema informa que el cliente no selecciono fecha y hora en que se realizaran los servicios.

Postcondición
Se realiza la factura como comprobante de la venta efectuada de los servicios de una orden de trabajo.


uc Caso de Uso
Usuario Cliente
CU - 013 – Facturación


class DER
TipoComprobante
«column»
*PK IdTipoComprobante: int
* Descripcion: nvarchar(50)
«PK»
+ PK_TipoComprobante(int)
ItemOrdenDeTrabajo
«column»
*PK IdItemOrdenDeTrabajo: int
FK IdOrdenDeTrabajo: int
*FK IdUsuario: int
* NombreUsuario: nvarchar(50)
*FK IdServicio: int
FK IdDireccion: int
Precio: decimal(18)
Fecha: date
Hora: time(7)
Cantidad: int
«PK»
+ PK_ItemOrdenDeTrabajo_1(int)
«FK»
+ FK_ItemOrdenDeTrabajo_Direccion(int)
+ FK_ItemOrdenDeTrabajo_OrdenDeTrabajo1(int)
+ FK_ItemOrdenDeTrabajo_Servicio1(int)
+ FK_ItemOrdenDeTrabajo_Usuario(int)
+PK_TipoComprobante
(IdTipoComprobante = IdTipoComprobante)
«FK»
+FK_Comprobante_TipoComprobante
Comprobante
«column»
*PK IdComprobante: int
*FK IdTipoComprobante: int
*FK IdSucursal: int
*FK IdOrdenDeTrabajo: int
* Fecha: date = (getdate())
«PK»
+ PK_Comprobante(int)
«FK»
+ FK_Comprobante_OrdenDeTrabajo(int)
+ FK_Comprobante_Sucursal(int)
+ FK_Comprobante_TipoComprobante(int)
+FK_ItemOrdenDeTrabajo_OrdenDeTrabajo1
(IdOrdenDeTrabajo = IdOrdenDeTrabajo)
«FK»
+FK_Comprobante_OrdenDeTrabajo
(IdOrdenDeTrabajo = IdOrdenDeTrabajo)
«FK»
+PK_OrdenDeTrabajo
+PK_OrdenDeTrabajo
OrdenDeTrabajo
«column»
*PK IdOrdenDeTrabajo: int
*FK IdUsuario: int
Fecha: datetime = (getdate())
FK IdEstado: int
«PK»
+ PK_OrdenDeTrabajo(int)
«FK»
+ FK_OrdenDeTrabajo_Estado1(int)
+ FK_OrdenDeTrabajo_Usuario1(int)

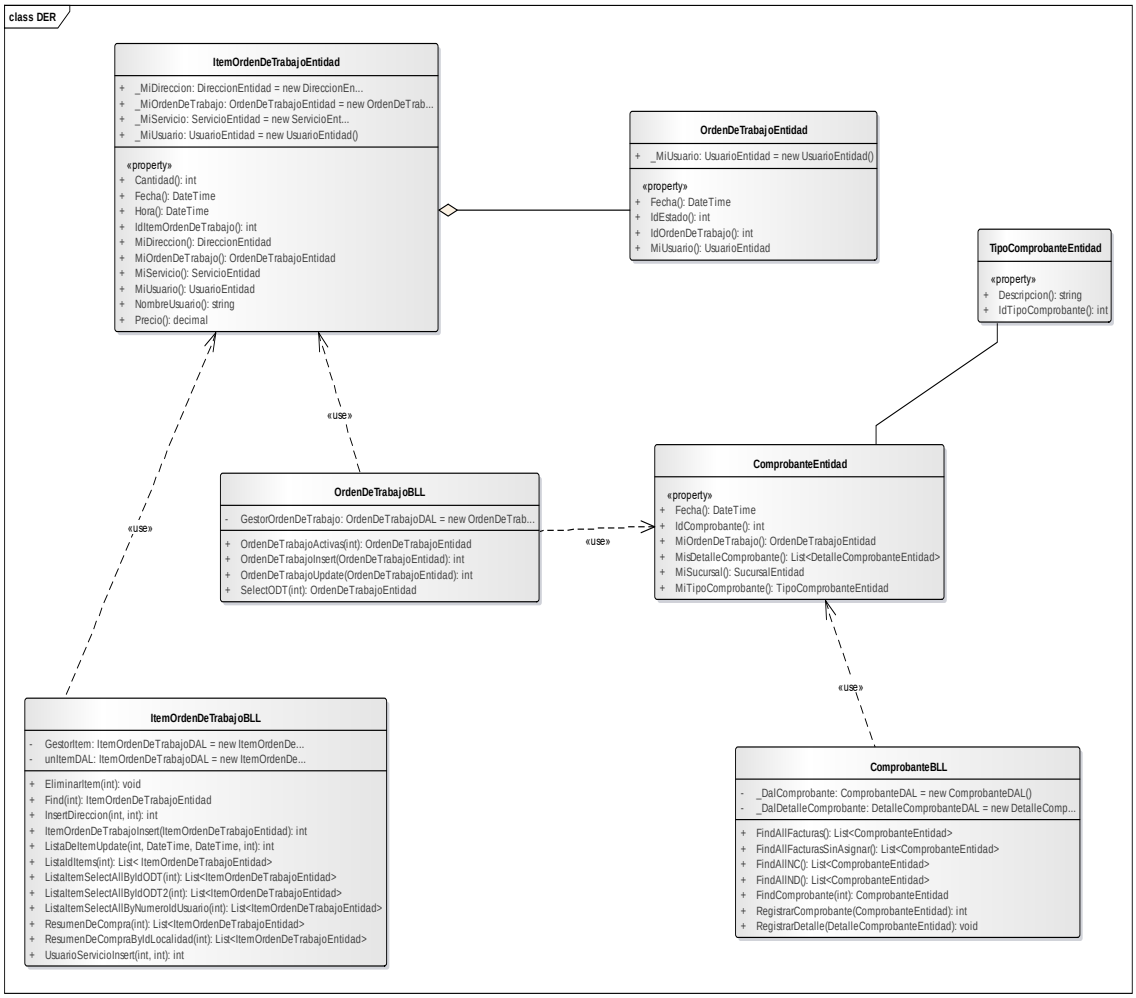
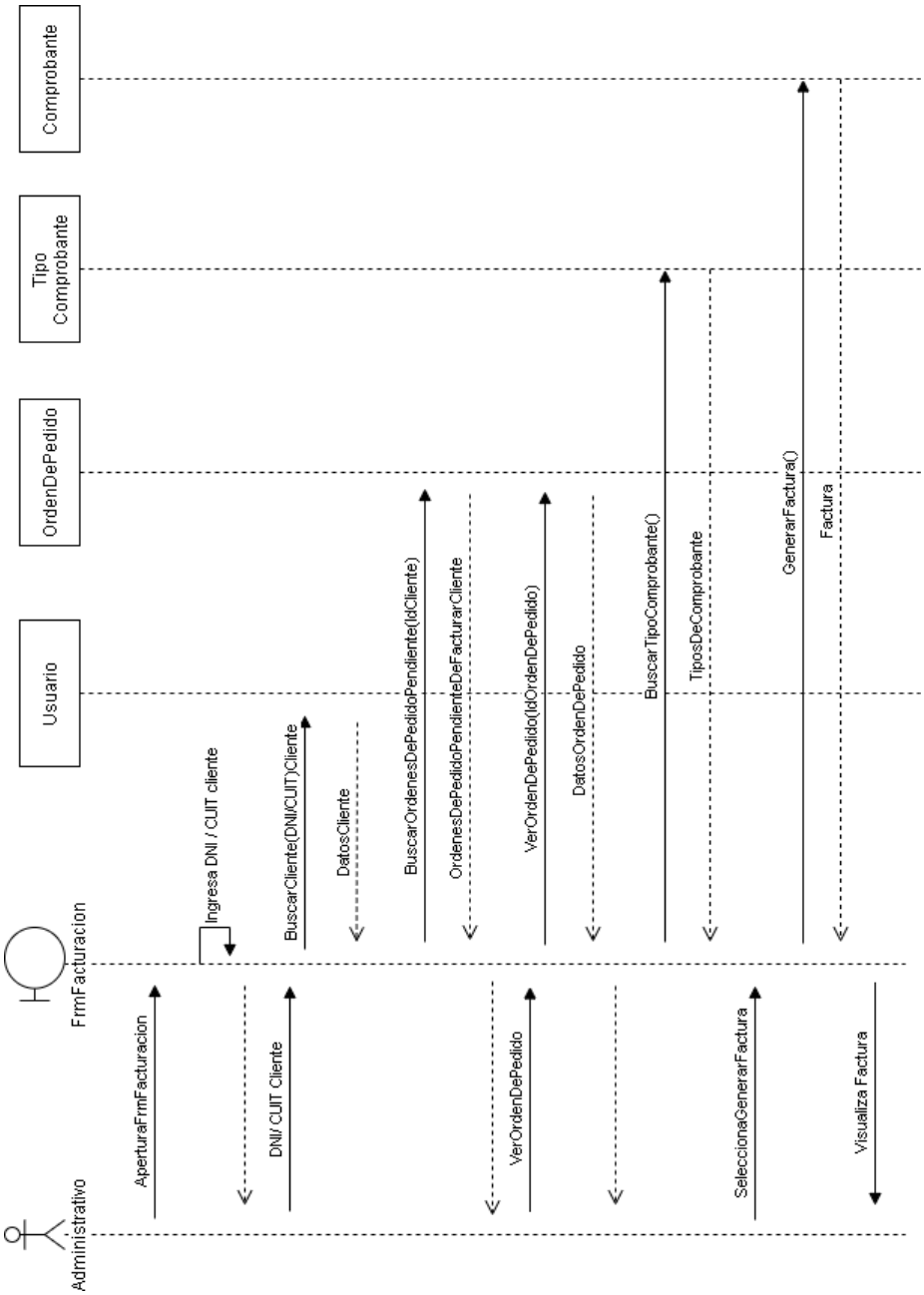
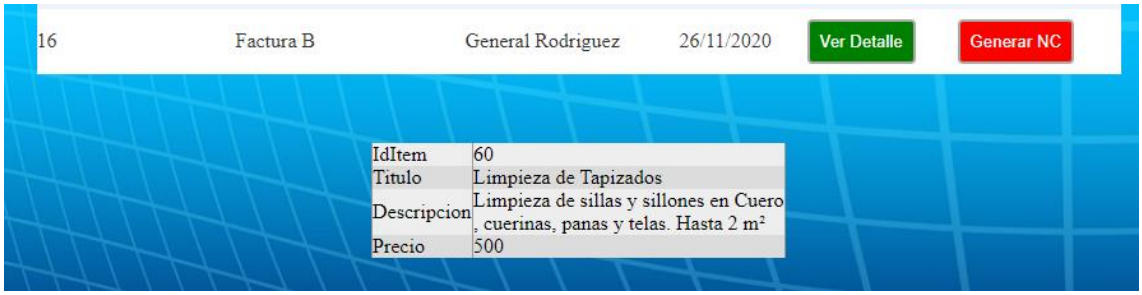

class DER
ItemOrdenDeTrabajoEntidad
+ _MiDireccion: DireccionEntidad = new DireccionEn...
+ _MiOrdenDeTrabajo: OrdenDeTrabajoEntidad = new OrdenDeTrab...
+ _MiServicio: ServicioEntidad = new ServicioEnt...
+ _MiUsuario: UsuarioEntidad = new UsuarioEntidad()
«property»
+ Cantidad(): int
+ Fecha(): DateTime
+ Hora(): DateTime
+ IdItemOrdenDeTrabajo(): int
+ MiDireccion(): DireccionEntidad
+ MiOrdenDeTrabajo(): OrdenDeTrabajoEntidad
+ MiServicio(): ServicioEntidad
+ MiUsuario(): UsuarioEntidad
+ NombreUsuario(): string
+ Precio(): decimal
OrdenDeTrabajoEntidad
+ _MiUsuario: UsuarioEntidad = new UsuarioEntidad()
«property»
+ Fecha(): DateTime
+ IdEstado(): int
+ IdOrdenDeTrabajo(): int
+ MiUsuario(): UsuarioEntidad
TipoComprobanteEntidad
«property»
+ Descripcion(): string
+ IdTipoComprobante(): int
«use»
«use»
OrdenDeTrabajoBLL
- GestorOrdenDeTrabajo: OrdenDeTrabajoDAL = new OrdenDeTrab...
+ OrdenDeTrabajoActivas(int): OrdenDeTrabajoEntidad
+ OrdenDeTrabajoInsert(OrdenDeTrabajoEntidad): int
+ OrdenDeTrabajoUpdate(OrdenDeTrabajoEntidad): int
+ SelectODT(int): OrdenDeTrabajoEntidad
«use»
ComprobanteEntidad
«property»
+ Fecha(): DateTime
+ IdComprobante(): int
+ MiOrdenDeTrabajo(): OrdenDeTrabajoEntidad
+ MisDetalleComprobante(): List<DetalleComprobanteEntidad>
+ MiSucursal(): SucursalEntidad
+ MiTipoComprobante(): TipoComprobanteEntidad
«use»
ItemOrdenDeTrabajoBLL
- GestorItem: ItemOrdenDeTrabajoDAL = new ItemOrdenDe...
- unItemDAL: ItemOrdenDeTrabajoDAL = new ItemOrdenDe...
+ EliminarItem(int): void
+ Find(int): ItemOrdenDeTrabajoEntidad
+ InsertDireccion(int, int): int
+ ItemOrdenDeTrabajoInsert(ItemOrdenDeTrabajoEntidad): int
+ ListaDeItemUpdate(int, DateTime, DateTime, int): int
+ ListaIdItems(int): List< ItemOrdenDeTrabajoEntidad>
+ ListaItemSelectAllByIdODT(int): List<ItemOrdenDeTrabajoEntidad>
+ ListaItemSelectAllByIdODT2(int): List<ItemOrdenDeTrabajoEntidad>
+ ListaItemSelectAllByNumeroIdUsuario(int): List<ItemOrdenDeTrabajoEntidad>
+ ResumenDeCompra(int): List<ItemOrdenDeTrabajoEntidad>
+ ResumenDeCompraByIdLocalidad(int): List<ItemOrdenDeTrabajoEntidad>
+ UsuarioServicioInsert(int, int): int
ComprobanteBLL
- _DalComprobante: ComprobanteDAL = new ComprobanteDAL()
- _DalDetalleComprobante: DetalleComprobanteDAL = new DetalleComp...
+ FindAllFacturas(): List<ComprobanteEntidad>
+ FindAllFacturasSinAsignar(): List<ComprobanteEntidad>
+ FindAllNC(): List<ComprobanteEntidad>
+ FindAllND(): List<ComprobanteEntidad>
+ FindComprobante(int): ComprobanteEntidad
+ RegistrarComprobante(ComprobanteEntidad): int
+ RegistrarDetalle(DetalleComprobanteEntidad): void

Diagrama de Secuencia CU - 013 – Facturación



Pantalla


16
Factura B
General Rodriguez
26/11/2020
Ver Detalle
Generar NC
IdItem 60
Titulo Limpieza de Tapizados
Descripcion Limpieza de sillas y sillones en Cuero, cuerinas, panas y telas. Hasta 2 m²
Precio 500

CU - 014– Emisión Nota de Crédito

Precondición
Debe haber una factura generada y que haya algún dato o monto incorrecto que haya que corregir en la misma o se deba realizar una acreditación al cliente por diversos motivos

Actores primarios:
Empleado Administrativo

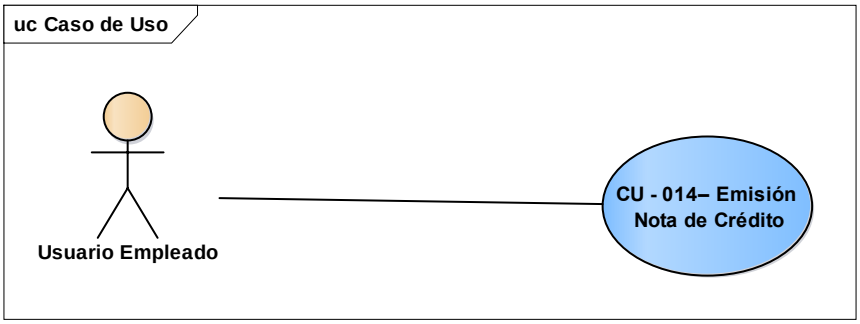
Camino principal del caso de uso – Emitir una nota de crédito

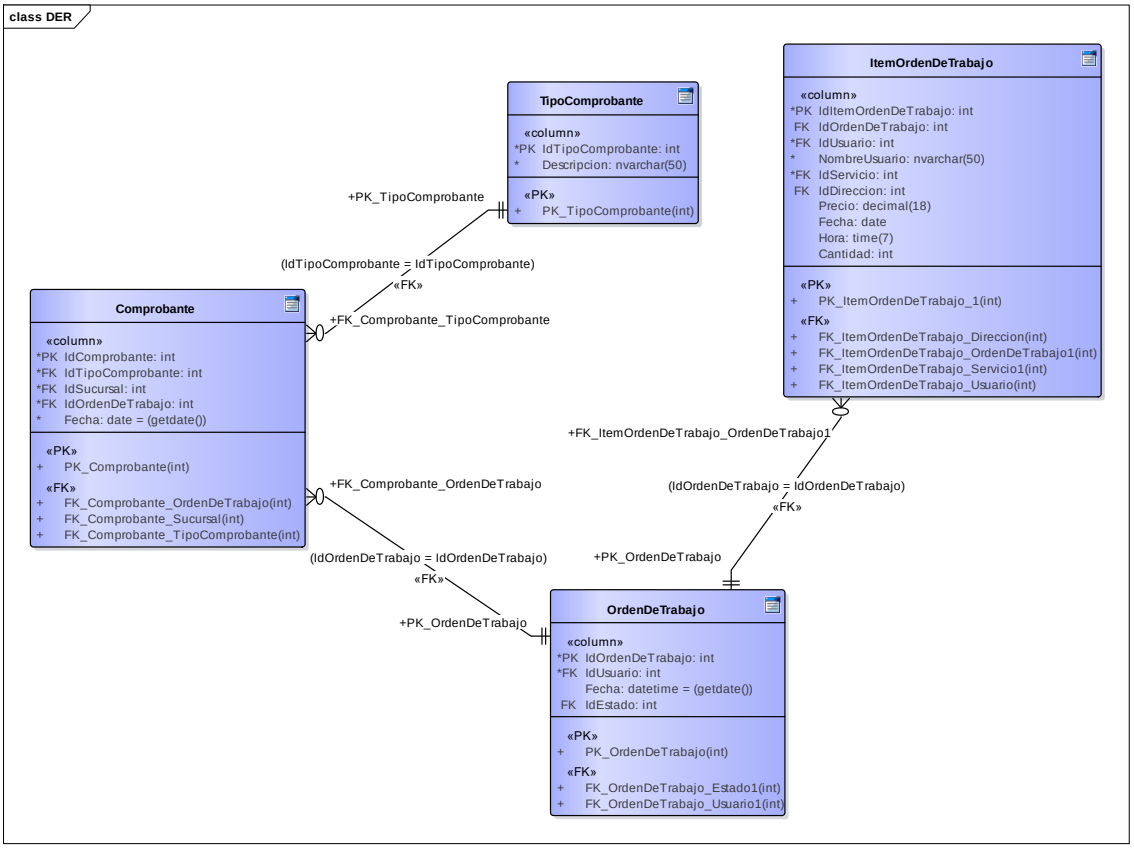
1. El administrativo se dirige a la sección de facturación
1. El administrativo selecciona la factura que está implicada en la emisión de la nota de crédito
2. El sistema busca la factura y muestra los datos de esta
    a. Fecha de emisión.
    b. Servicios, cantidades, precios unitarios, precio total final.
    c. Nro. de Orden de trabajo asociada.

3. El administrativo selecciona la opción para generar una nota de crédito.
4. El sistema muestra la genera de nota de crédito anulando en su totalidad a la factura por orden de trabajo.

5. El sistema emite la nota de crédito asociada a la factura pertinente con los datos:
    a. Fecha de emisión.
    b. Monto acreditado.
    c. Nro. de Orden de trabajo.

Postcondición:
Se emite una Nota de crédito


uc Caso de Uso
Usuario Empleado
CU - 014– Emisión Nota de Crédito


class DER
TipoComprobante
«column»
*PK IdTipoComprobante: int
* Descripcion: nvarchar(50)
«PK»
+ PK_TipoComprobante(int)
ItemOrdenDeTrabajo
«column»
*PK IdItemOrdenDeTrabajo: int
FK IdOrdenDeTrabajo: int
*FK IdUsuario: int
* NombreUsuario: nvarchar(50)
*FK IdServicio: int
FK IdDireccion: int
Precio: decimal(18)
Fecha: date
Hora: time(7)
Cantidad: int
«PK»
+ PK_ItemOrdenDeTrabajo_1(int)
«FK»
+ FK_ItemOrdenDeTrabajo_Direccion(int)
+ FK_ItemOrdenDeTrabajo_OrdenDeTrabajo1(int)
+ FK_ItemOrdenDeTrabajo_Servicio1(int)
+ FK_ItemOrdenDeTrabajo_Usuario(int)
+PK_TipoComprobante
(IdTipoComprobante = IdTipoComprobante)
«FK»
+FK_Comprobante_TipoComprobante
Comprobante
«column»
*PK IdComprobante: int
*FK IdTipoComprobante: int
*FK IdSucursal: int
*FK IdOrdenDeTrabajo: int
* Fecha: date = (getdate())
«PK»
+ PK_Comprobante(int)
«FK»
+ FK_Comprobante_OrdenDeTrabajo(int)
+ FK_Comprobante_Sucursal(int)
+ FK_Comprobante_TipoComprobante(int)
+FK_ItemOrdenDeTrabajo_OrdenDeTrabajo1
(IdOrdenDeTrabajo = IdOrdenDeTrabajo)
«FK»
+FK_Comprobante_OrdenDeTrabajo
(IdOrdenDeTrabajo = IdOrdenDeTrabajo)
«FK»
+PK_OrdenDeTrabajo
+PK_OrdenDeTrabajo
OrdenDeTrabajo
«column»
*PK IdOrdenDeTrabajo: int
*FK IdUsuario: int
Fecha: datetime = (getdate())
FK IdEstado: int
«PK»
+ PK_OrdenDeTrabajo(int)
«FK»
+ FK_OrdenDeTrabajo_Estado1(int)
+ FK_OrdenDeTrabajo_Usuario1(int)

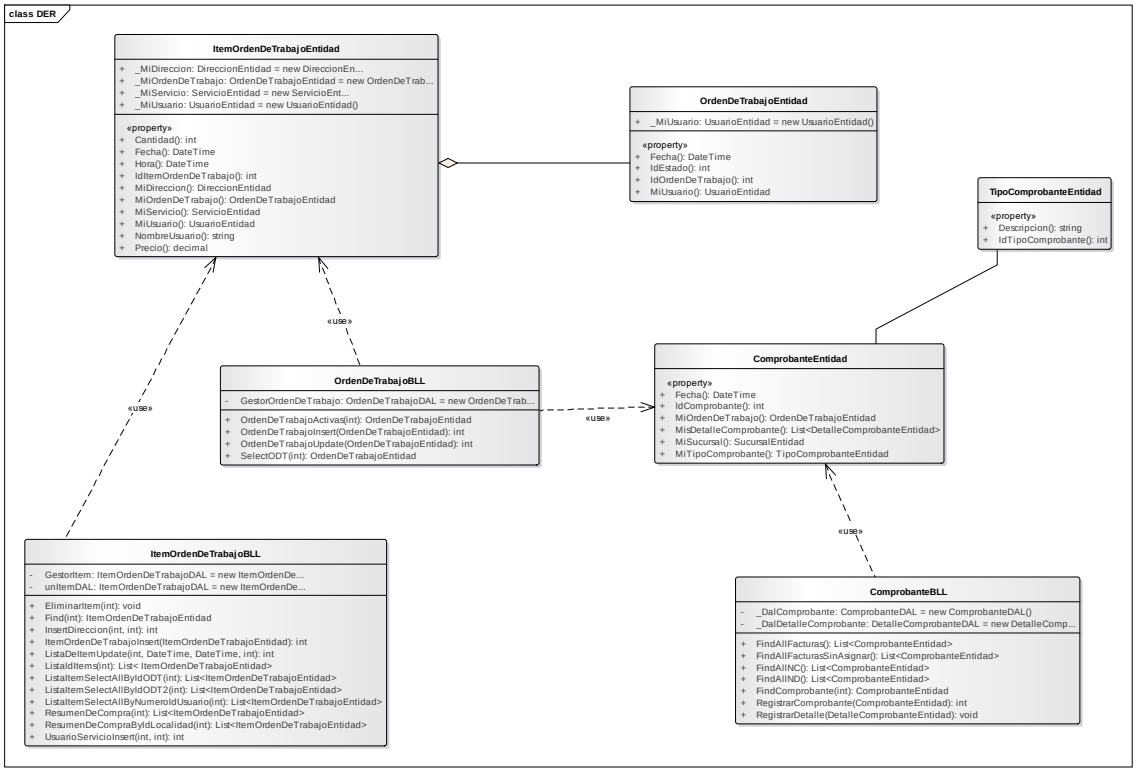
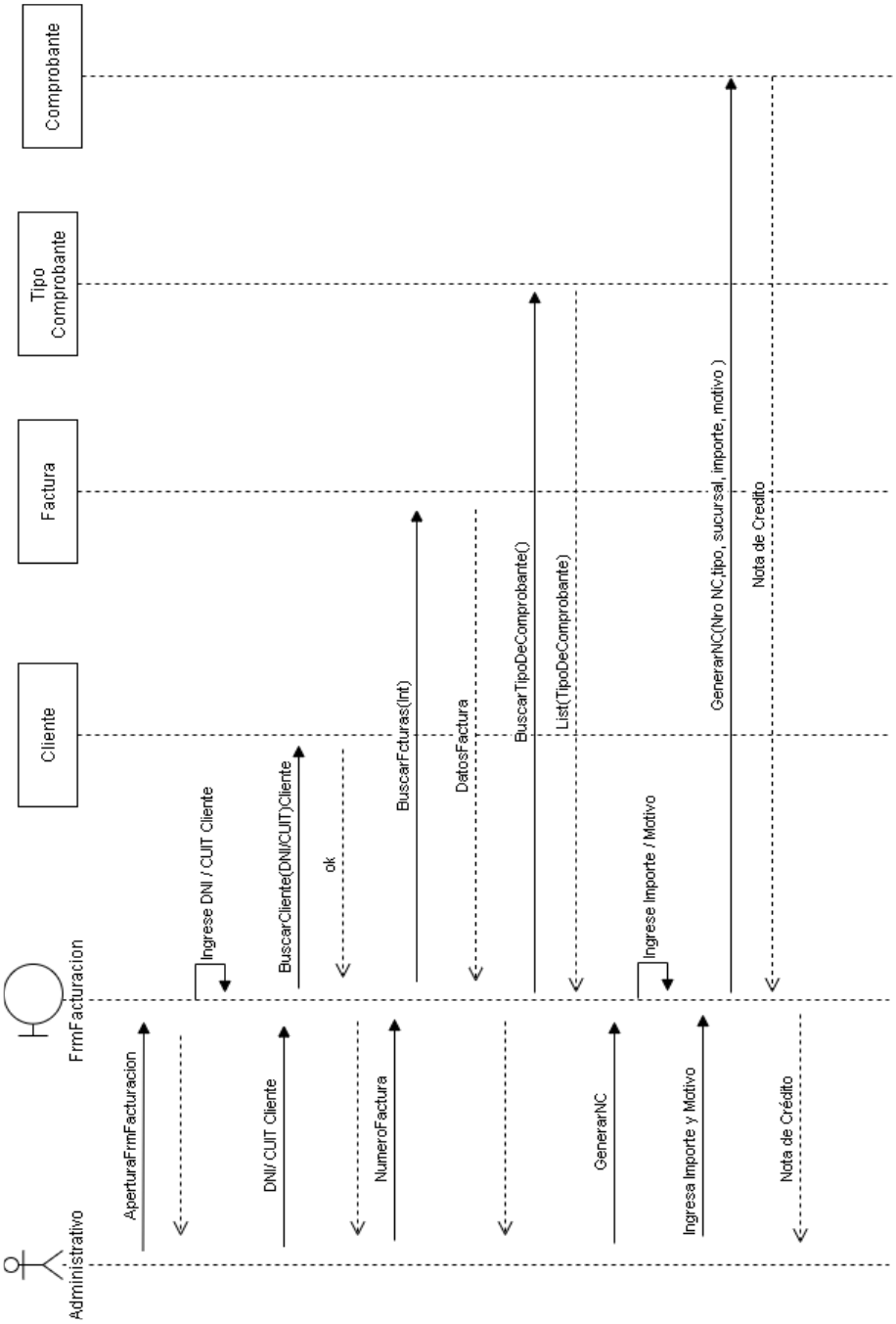
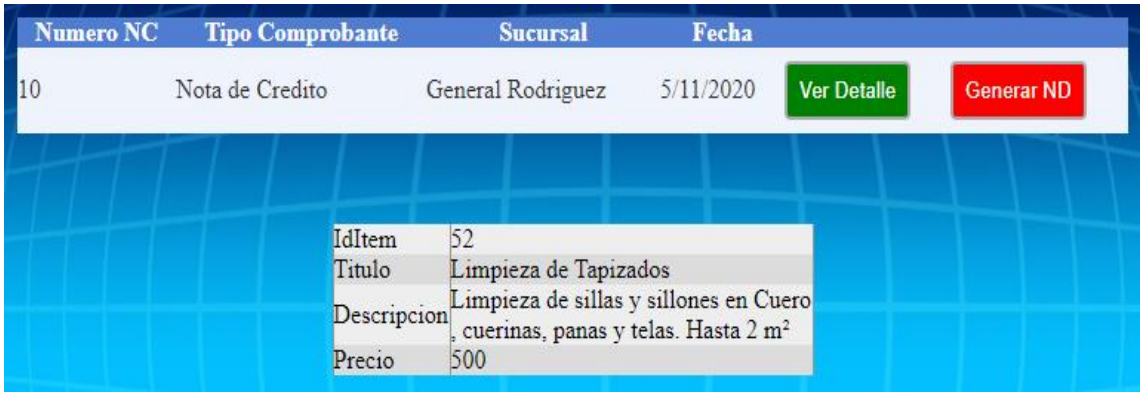


Diagrama de Secuencia CU - 014 – Emisión Nota de Crédito


Administrativo
FrmFacturacion
Cliente
Factura
Tipo Comprobante
Comprobante
AperturaFrmFacturacion
Ingrese DNI / CUIT Cliente
DNI/ CUIT Cliente
BuscarCliente(DNI/CUIT):Cliente
ok
NumeroFactura
BuscarFcturas(int)
DatosFactura
GenerarNC
BuscarTipoDeComprobante()
List(TipoDeComprobante)
Ingresa Importe y Motivo
Ingrese Importe / Motivo
GenerarNC(Nro NC,tipo, sucursal, importe, motivo)
Nota de Crédito
Nota de Credito

Pantalla


Numero NC
Tipo Comprobante
Sucursal
Fecha
10
Nota de Credito
General Rodriguez
5/11/2020
Ver Detalle
Generar ND
IdItem
52
Titulo
Limpieza de Tapizados
Descripcion
Limpieza de sillas y sillones en Cuero , cuerinas, panas y telas. Hasta 2 m²
Precio
500

CU - 015 – Emisión de Notas de Debito

Precondición
Debe haber una factura generada y que haya algún dato o monto incorrecto que haya que corregir en la misma o se deba realizar un debito al cliente por diversos motivos.

Actores primarios:
Empleado Administrativo

Camino principal del caso de uso – Emisión de nota de Debito

1. El administrativo se dirige a la sección de Nota de Debito
3. El sistema busca los comprobantes disponibles para aplicar nota de débito y muestra los datos de esta

    a. Numero de comprobante.
    b. Sucursal.
    c. Fecha de emisión.
    d. Servicios, cantidades, precios unitarios.
    e. Nro. de Orden de trabajo asociada.

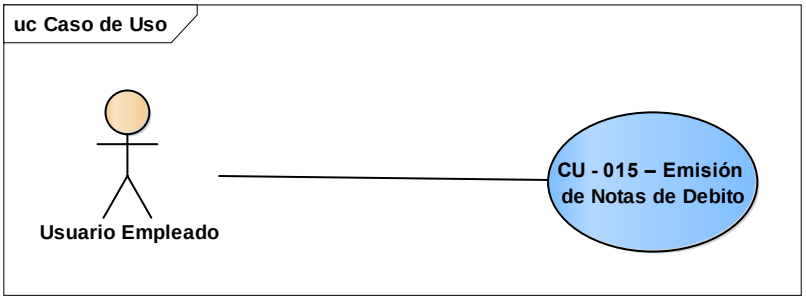
4. El administrativo selecciona la opción para generar una nota de débito.

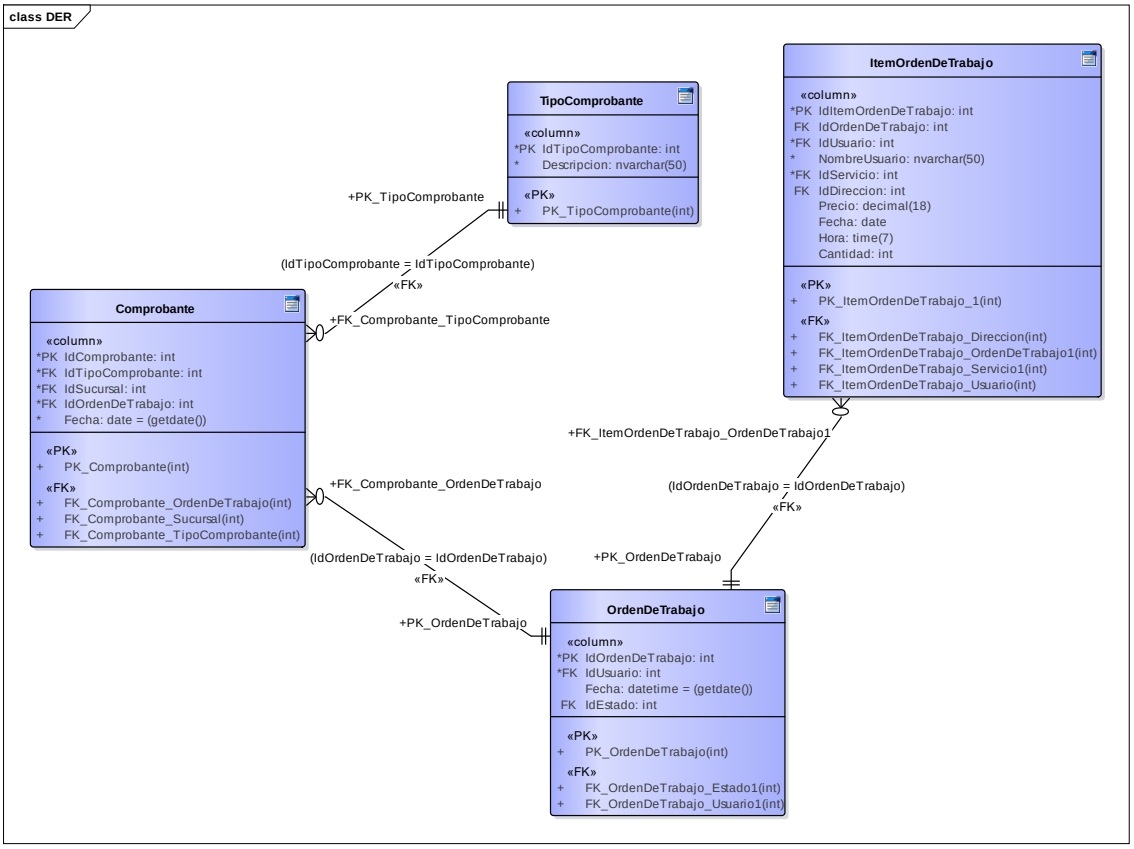
5. El sistema emite la nota de débito asociada a la orden de trabajo pertinente con los datos:

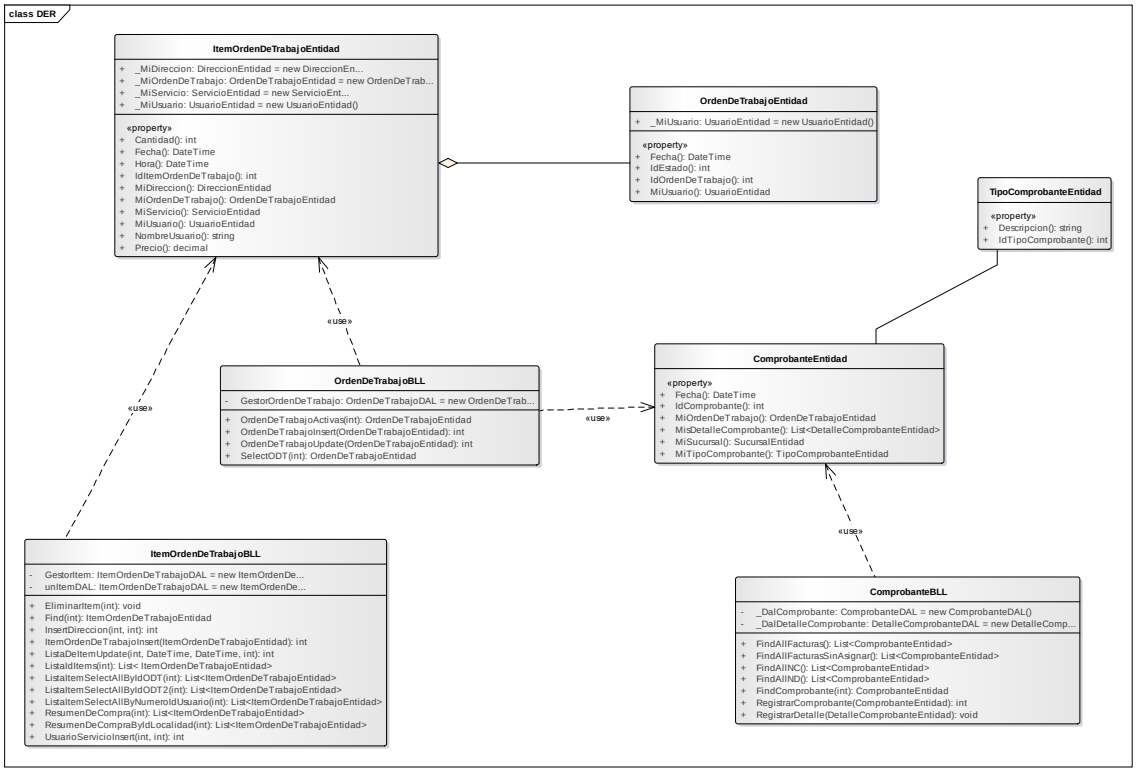
    a. Numero de comprobante.
    b. Sucursal.
    c. Fecha de emisión.
    d. Servicios, cantidades, precios unitarios.
    e. Nro. de Orden de trabajo asociada.

6. El sistema muestra la Nota de Debito generada.

Postcondición:
Se emite una Nota de débito.


uc Caso de Uso
Usuario Empleado
CU - 015 – Emisión de Notas de Debito


class DER
TipoComprobante
«column»
*PK IdTipoComprobante: int
* Descripcion: nvarchar(50)
«PK»
+ PK_TipoComprobante(int)
ItemOrdenDeTrabajo
«column»
*PK IdItemOrdenDeTrabajo: int
FK IdOrdenDeTrabajo: int
*FK IdUsuario: int
* NombreUsuario: nvarchar(50)
*FK IdServicio: int
FK IdDireccion: int
Precio: decimal(18)
Fecha: date
Hora: time(7)
Cantidad: int
«PK»
+ PK_ItemOrdenDeTrabajo_1(int)
«FK»
+ FK_ItemOrdenDeTrabajo_Direccion(int)
+ FK_ItemOrdenDeTrabajo_OrdenDeTrabajo1(int)
+ FK_ItemOrdenDeTrabajo_Servicio1(int)
+ FK_ItemOrdenDeTrabajo_Usuario(int)
+PK_TipoComprobante
(IdTipoComprobante = IdTipoComprobante)
«FK»
+FK_Comprobante_TipoComprobante
Comprobante
«column»
*PK IdComprobante: int
*FK IdTipoComprobante: int
*FK IdSucursal: int
*FK IdOrdenDeTrabajo: int
* Fecha: date = (getdate())
«PK»
+ PK_Comprobante(int)
«FK»
+ FK_Comprobante_OrdenDeTrabajo(int)
+ FK_Comprobante_Sucursal(int)
+ FK_Comprobante_TipoComprobante(int)
+FK_ItemOrdenDeTrabajo_OrdenDeTrabajo1
(IdOrdenDeTrabajo = IdOrdenDeTrabajo)
«FK»
+FK_Comprobante_OrdenDeTrabajo
(IdOrdenDeTrabajo = IdOrdenDeTrabajo)
«FK»
+PK_OrdenDeTrabajo
+PK_OrdenDeTrabajo
OrdenDeTrabajo
«column»
*PK IdOrdenDeTrabajo: int
*FK IdUsuario: int
Fecha: datetime = (getdate())
FK IdEstado: int
«PK»
+ PK_OrdenDeTrabajo(int)
«FK»
+ FK_OrdenDeTrabajo_Estado1(int)
+ FK_OrdenDeTrabajo_Usuario1(int)


class DER
ItemOrdenDeTrabajoEntidad
+ _MiDireccion: DireccionEntidad = new DireccionEn...
+ _MiOrdenDeTrabajo: OrdenDeTrabajoEntidad = new OrdenDeTrab...
+ _MiServicio: ServicioEntidad = new ServicioEnt...
+ _MiUsuario: UsuarioEntidad = new UsuarioEntidad()
«property»
+ Cantidad(): int
+ Fecha(): DateTime
+ Hora(): DateTime
+ IdItemOrdenDeTrabajo(): int
+ MiDireccion(): DireccionEntidad
+ MiOrdenDeTrabajo(): OrdenDeTrabajoEntidad
+ MiServicio(): ServicioEntidad
+ MiUsuario(): UsuarioEntidad
+ NombreUsuario(): string
+ Precio(): decimal
OrdenDeTrabajoEntidad
+ _MiUsuario: UsuarioEntidad = new UsuarioEntidad()
«property»
+ Fecha(): DateTime
+ IdEstado(): int
+ IdOrdenDeTrabajo(): int
+ MiUsuario(): UsuarioEntidad
TipoComprobanteEntidad
«property»
+ Descripcion(): string
+ IdTipoComprobante(): int
«use»
OrdenDeTrabajoBLL
- GestorOrdenDeTrabajo: OrdenDeTrabajoDAL = new OrdenDeTrab...
+ OrdenDeTrabajoActivas(int): OrdenDeTrabajoEntidad
+ OrdenDeTrabajoInsert(OrdenDeTrabajoEntidad): int
+ OrdenDeTrabajoUpdate(OrdenDeTrabajoEntidad): int
+ SelectODT(int): OrdenDeTrabajoEntidad
«use»
ComprobanteEntidad
«property»
+ Fecha(): DateTime
+ IdComprobante(): int
+ MiOrdenDeTrabajo(): OrdenDeTrabajoEntidad
+ MisDetalleComprobante(): List<DetalleComprobanteEntidad>
+ MiSucursal(): SucursalEntidad
+ MiTipoComprobante(): TipoComprobanteEntidad
«use»
«use»
ItemOrdenDeTrabajoBLL
- GestorItem: ItemOrdenDeTrabajoDAL = new ItemOrdenDe...
- unItemDAL: ItemOrdenDeTrabajoDAL = new ItemOrdenDe...
+ EliminarItem(int): void
+ Find(int): ItemOrdenDeTrabajoEntidad
+ InsertDireccion(int, int): int
+ ItemOrdenDeTrabajoInsert(ItemOrdenDeTrabajoEntidad): int
+ ListaDeItemUpdate(int, DateTime, DateTime, int): int
+ ListaIdItems(int): List< ItemOrdenDeTrabajoEntidad>
+ ListaItemSelectAllByIdODT(int): List<ItemOrdenDeTrabajoEntidad>
+ ListaItemSelectAllByIdODT2(int): List<ItemOrdenDeTrabajoEntidad>
+ ListaItemSelectAllByNumeroIdUsuario(int): List<ItemOrdenDeTrabajoEntidad>
+ ResumenDeCompra(int): List<ItemOrdenDeTrabajoEntidad>
+ ResumenDeCompraByIdLocalidad(int): List<ItemOrdenDeTrabajoEntidad>
+ UsuarioServicioInsert(int, int): int
ComprobanteBLL
- _DalComprobante: ComprobanteDAL = new ComprobanteDAL()
- _DalDetalleComprobante: DetalleComprobanteDAL = new DetalleComp...
+ FindAllFacturas(): List<ComprobanteEntidad>
+ FindAllFacturasSinAsignar(): List<ComprobanteEntidad>
+ FindAllNC(): List<ComprobanteEntidad>
+ FindAllND(): List<ComprobanteEntidad>
+ FindComprobante(int): ComprobanteEntidad
+ RegistrarComprobante(ComprobanteEntidad): int
+ RegistrarDetalle(DetalleComprobanteEntidad): void

Pantalla

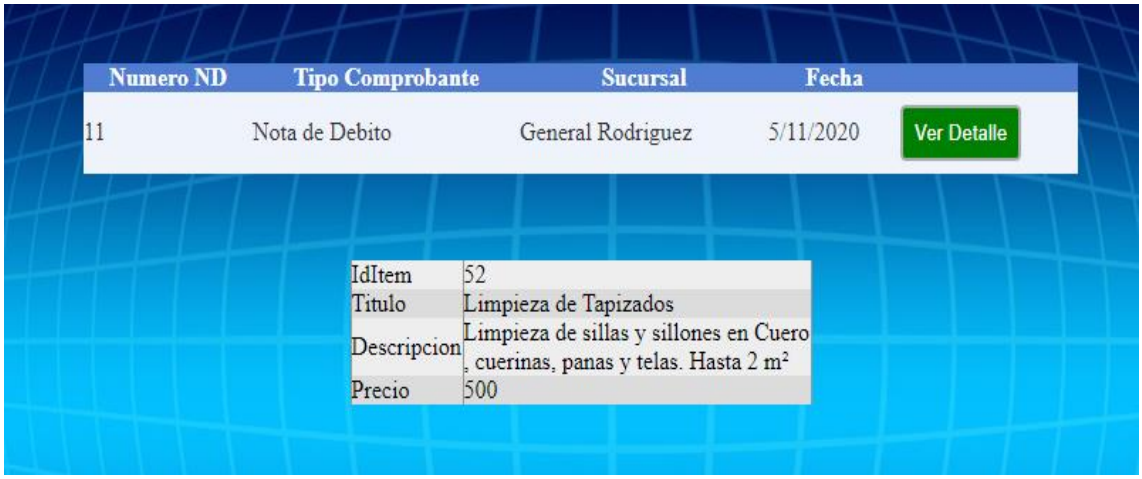
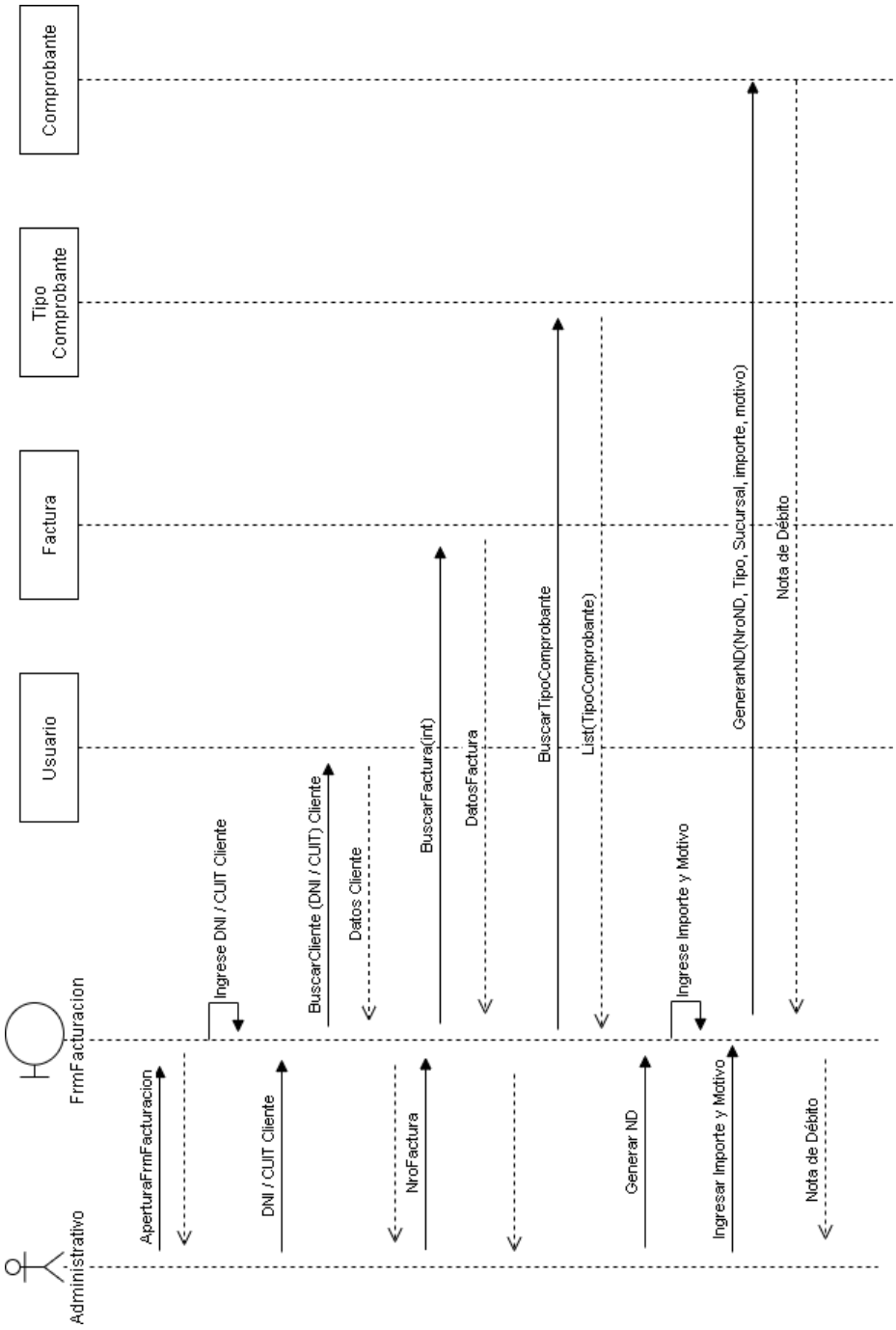

Numero ND
Tipo Comprobante
Sucursal
Fecha
11
Nota de Debito
General Rodriguez
5/11/2020
Ver Detalle
IdItem
52
Titulo
Limpieza de Tapizados
Descripcion
Limpieza de sillas y sillones en Cuero, cuerinas, panas y telas. Hasta 2 m²
Precio
500

Diagrama de Secuencia CU - 015 – Emisión de Notas de Debito



CU - 016 - Cobranza

Precondición:
No existen precondiciones

Actores primarios:
Cliente

Camino principal del caso de uso - Realizar un pago de Servicio

1. El Cliente se dirige a pagar el pedido
2. El Sistema presenta en pantalla los medios de pago disponibles:
    a. Transferencia bancaria
    b. Tarjeta de crédito
3. El Cliente selecciona la forma de pago “Tarjeta de crédito”
4. El Sistema muestra los datos de la tarjeta que posee el Cliente en forma predeterminada y solicita confirmación del pago:
    a. Titular de tarjeta
    b. Tipo de tarjeta
    c. Número de tarjeta
    d. Fecha Expiración de tarjeta
    e. Opción “Registrar / Actualizar la Tarjeta” para cambiar de tarjeta
    f. Opción “Pagar” para pagar
5. El Cliente selecciona la opción “Pagar”
6. El Sistema realiza el pago contra el proveedor de la tarjeta de crédito, registra una factura, coloca el Pedido en estado “Finalizado” y muestra el mensaje “Felicitaciones, Pago Exitoso”
7. El sistema genera Comprobante de pago “Recibo”. CU – O44 – Generar Recibo
8. El Cliente acepta el mensaje

Caminos Alternativos

Para el paso 3: El Cliente selecciona la forma de pago “Transferencia bancaria”

1. El Sistema envía un mail al cliente con los datos para la transferencia y emite un mensaje indicando “El correo se ha enviado con los datos de transferencia”

Para el paso 4: El Cliente no posee tarjetas de crédito registradas

1. El Sistema muestra un mensaje emergente indicando “Usted no posee tarjeta registrada. Registre una para continuar el proceso” y ofrece la opción “Elegir otra tarjeta” para agregar tarjetas

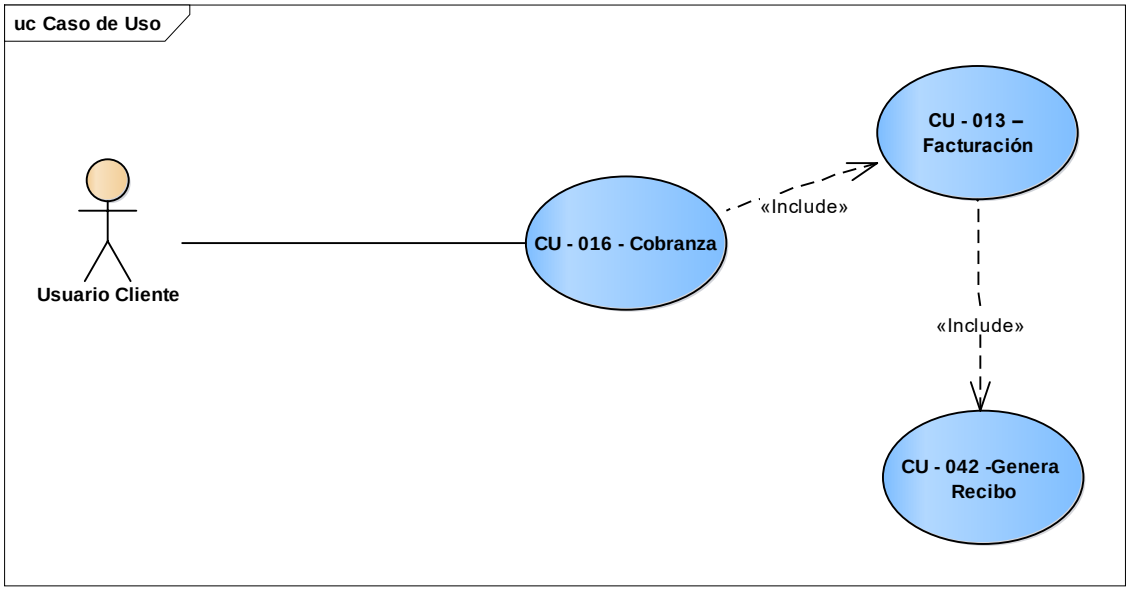
Para el paso 5: El cliente paga con tarjeta de debito

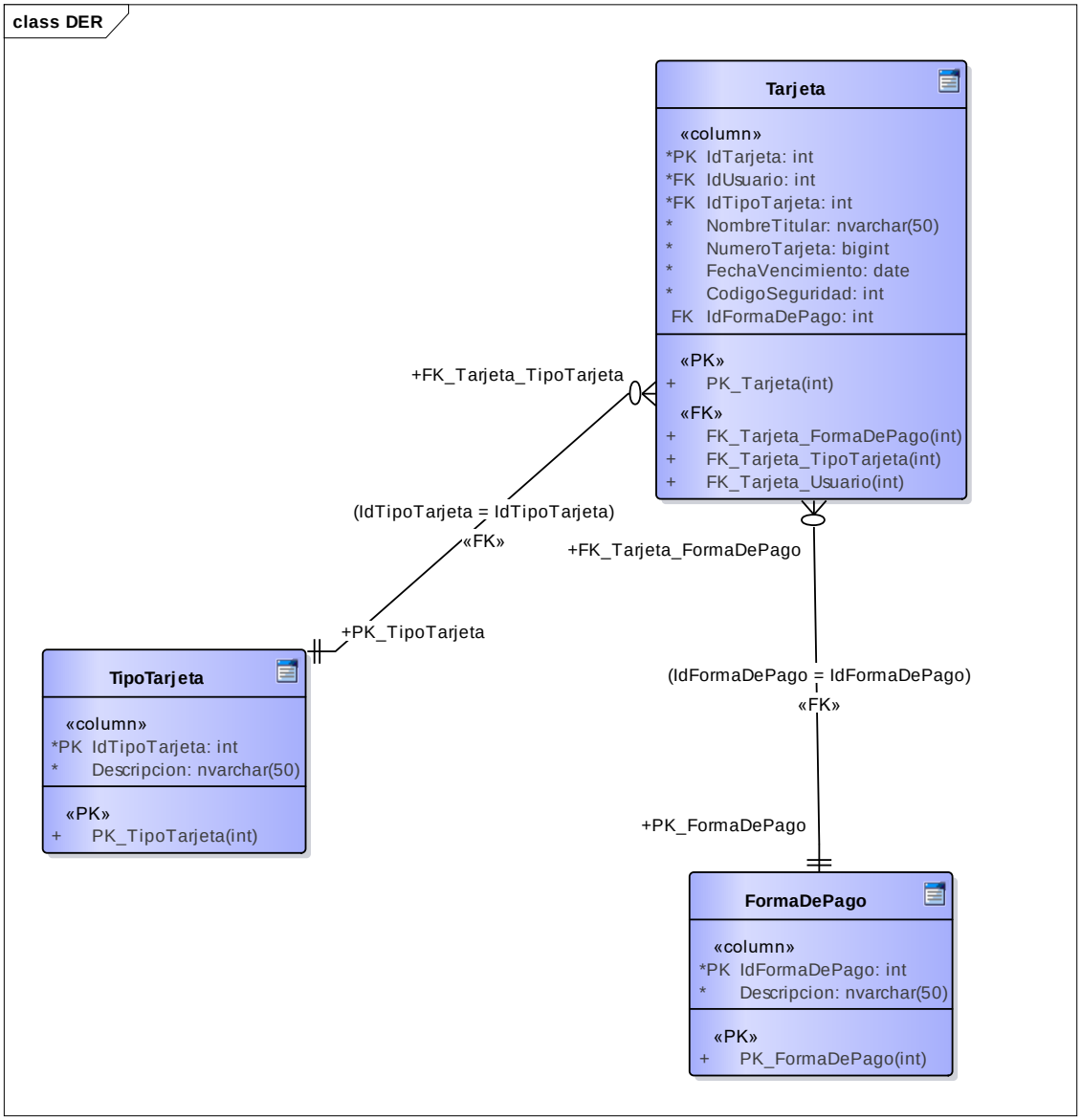
5 a. Pasa la tarjeta a través del servicio posnet en la Oficina.
5 b. El sistema solicita que deslice la tarjeta de débito por la ranura.
5.c. El administrativo ingresa cantidad de cuotas.
5 d. El sistema muestra todos los datos ingresados y solicita se ingresen los datos de la tarjeta de débito:

a. Código de seguridad: 4 números máximo

5 e. El administrativo ingresa los datos de la tarjeta de débito y confirma la compra y el curso sigue en el punto 7.

Postcondición:
Emisión de Factura.
Emisión de recibo.


uc Caso de Uso
Usuario Cliente
CU - 016 - Cobranza
«Include»
CU - 013 – Facturación
«Include»
CU - 042 -Genera Recibo


class DER
Tarjeta
«column»
*PK IdTarjeta: int
*FK IdUsuario: int
*FK IdTipoTarjeta: int
* NombreTitular: nvarchar(50)
* NumeroTarjeta: bigint
* FechaVencimiento: date
* CodigoSeguridad: int
FK IdFormaDePago: int
«PK»
+ PK_Tarjeta(int)
«FK»
+ FK_Tarjeta_FormaDePago(int)
+ FK_Tarjeta_TipoTarjeta(int)
+ FK_Tarjeta_Usuario(int)
+FK_Tarjeta_TipoTarjeta
(IdTipoTarjeta = IdTipoTarjeta)
«FK»
+FK_Tarjeta_FormaDePago
+PK_TipoTarjeta
TipoTarjeta
«column»
*PK IdTipoTarjeta: int
* Descripcion: nvarchar(50)
«PK»
+ PK_TipoTarjeta(int)
(IdFormaDePago = IdFormaDePago)
«FK»
+PK_FormaDePago
FormaDePago
«column»
*PK IdFormaDePago: int
* Descripcion: nvarchar(50)
«PK»
+ PK_FormaDePago(int)

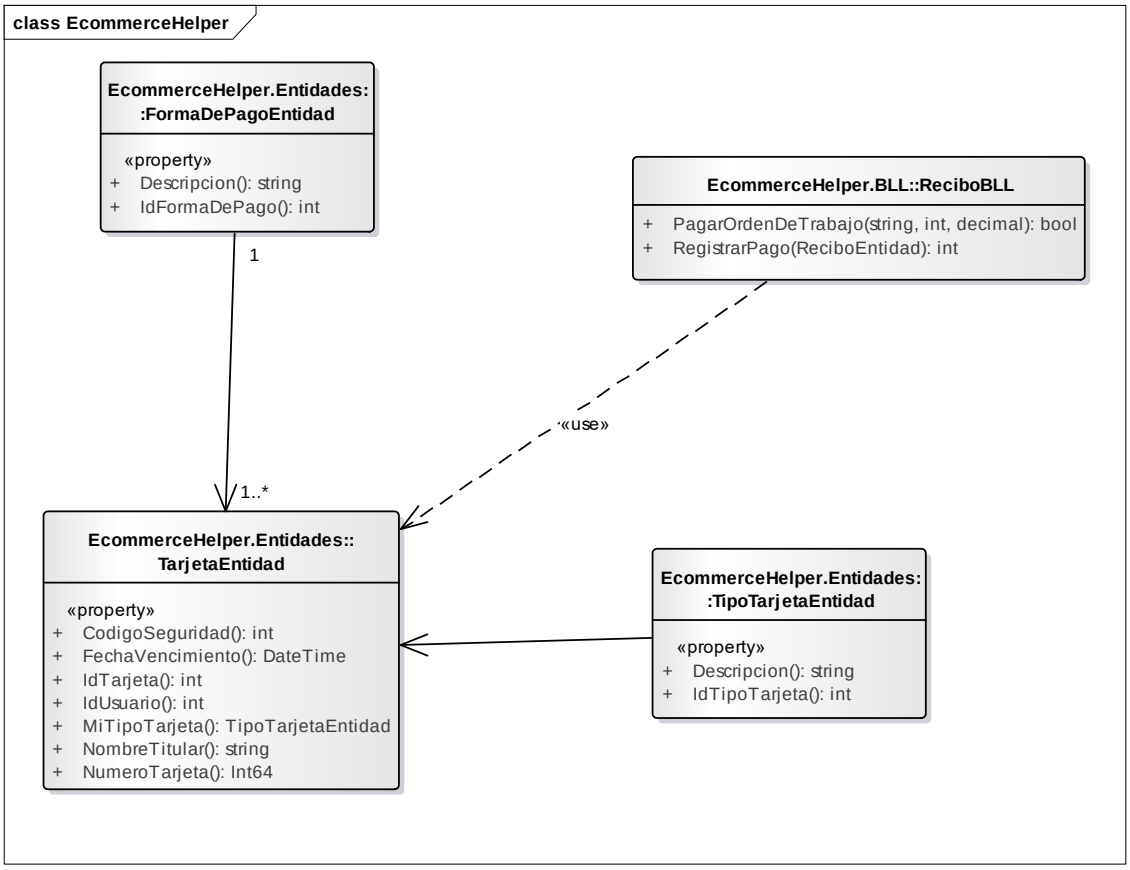
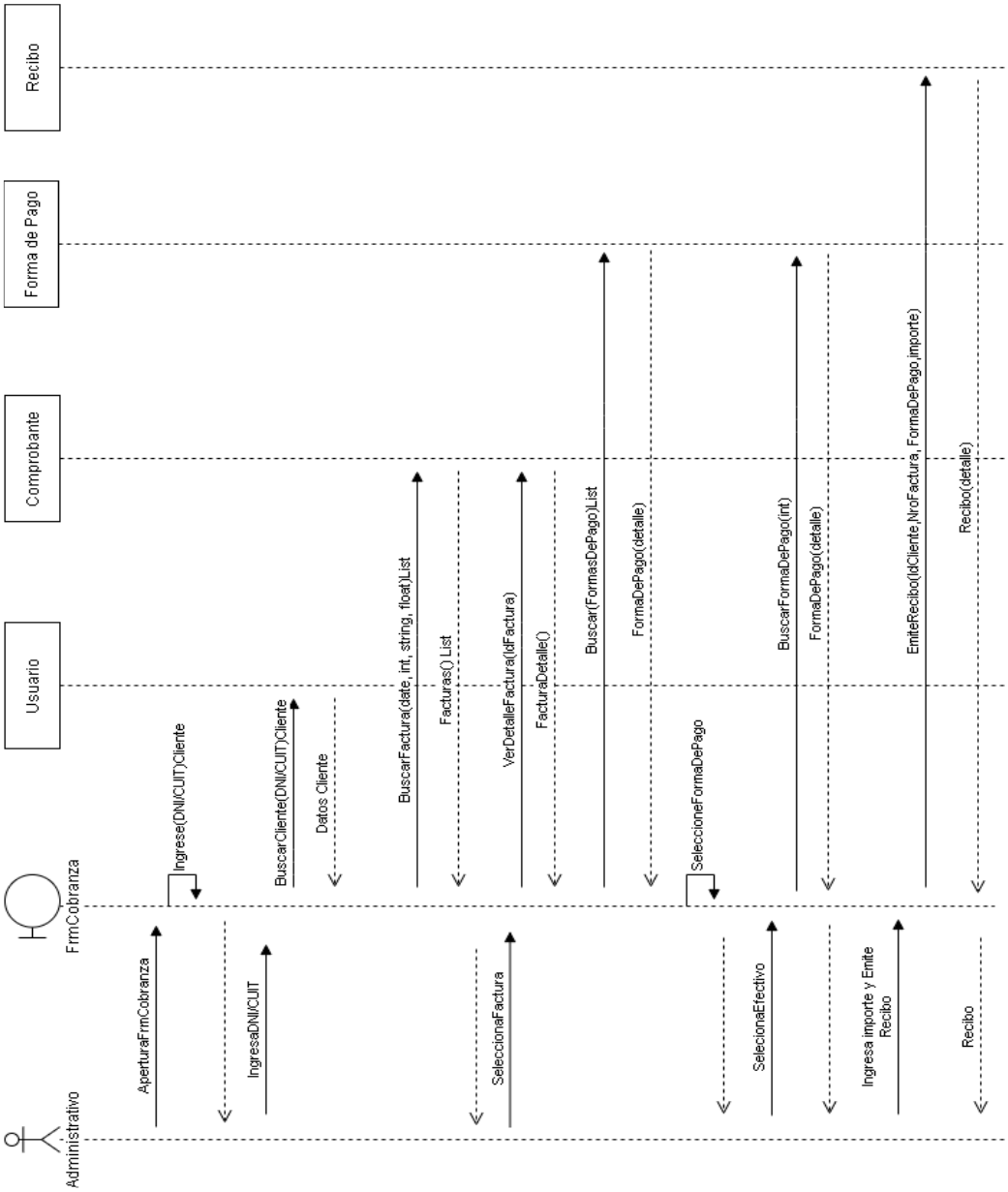

class EcommerceHelper
EcommerceHelper.Entidades::FormaDePagoEntidad
«property»
+ Descripcion(): string
+ IdFormaDePago(): int
1
1..*
EcommerceHelper.BLL::ReciboBLL
+ PagarOrdenDeTrabajo(string, int, decimal): bool
+ RegistrarPago(ReciboEntidad): int
«use»
EcommerceHelper.Entidades::TarjetaEntidad
«property»
+ CodigoSeguridad(): int
+ FechaVencimiento(): DateTime
+ IdTarjeta(): int
+ IdUsuario(): int
+ MiTipoTarjeta(): TipoTarjetaEntidad
+ NombreTitular(): string
+ NumeroTarjeta(): Int64
EcommerceHelper.Entidades::TipoTarjetaEntidad
«property»
+ Descripcion(): string
+ IdTipoTarjeta(): int

Diagrama de Secuencia CU - 016 – Cobranza



Pantalla

Titular	Numero	Tipo Tarjeta	Fecha de Vencimiento	Codigo de Seguridad	
Esteban Ensueño Pesadilla	4398189693448976	Credito	30/11/2020	111	Seleccionar

Registrar / Actualizar Tarjeta

CU - 044 -Genera Recibo

Precondición
Debe haber una orden de trabajo.

Actores primarios:
Cliente

Camino principal del caso de uso – Genera Recibo

1. CU - 016 – Cobranza

2. El sistema genera Comprobante de pago donde queda registrado el número que identifica la orden de trabajo, fecha, importe y el tipo de pago que se utilizó.

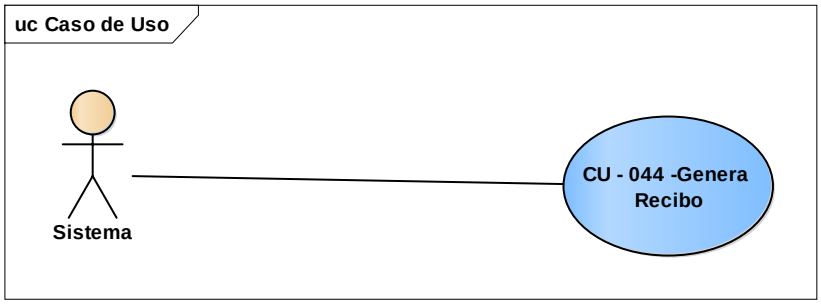
3 El Cliente acepta el mensaje

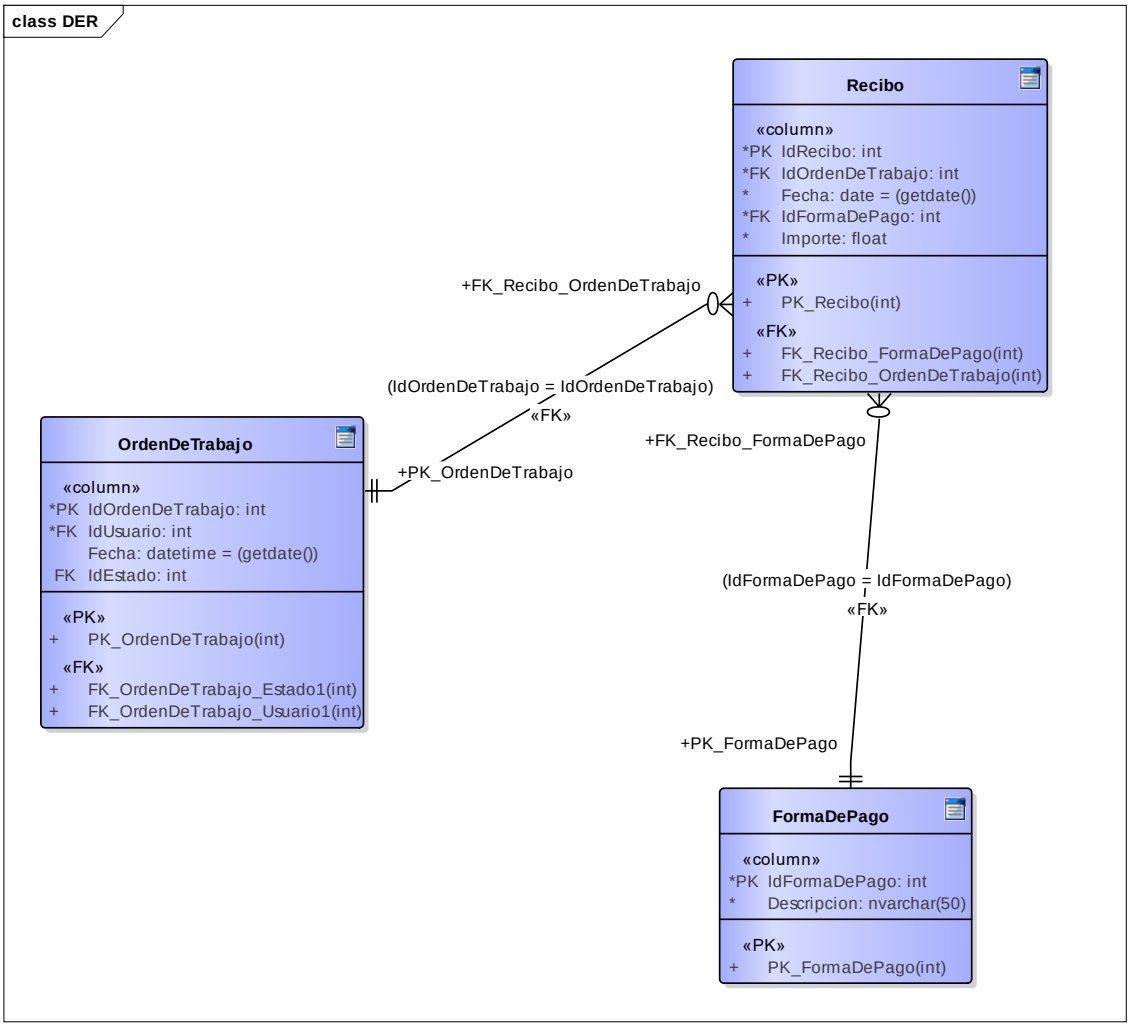
Caminos Alternativos

Para el paso 3: medio de pago rechazado

3 1. El sistema informa que el cliente medio de pago por el cual realizo el abono de la factura fue rechazado

Postcondición
Se registra el comprobante de pago Recibo en la base de datos.


uc Caso de Uso
Sistema
CU - 044 -Genera Recibo


class DER
Recibo
«column»
*PK IdRecibo: int
*FK IdOrdenDeTrabajo: int
* Fecha: date = (getdate())
*FK IdFormaDePago: int
* Importe: float
«PK»
+ PK_Recibo(int)
«FK»
+ FK_Recibo_FormaDePago(int)
+ FK_Recibo_OrdenDeTrabajo(int)
+FK_Recibo_OrdenDeTrabajo
(IdOrdenDeTrabajo = IdOrdenDeTrabajo)
«FK»
+PK_OrdenDeTrabajo
+FK_Recibo_FormaDePago
OrdenDeTrabajo
«column»
*PK IdOrdenDeTrabajo: int
*FK IdUsuario: int
Fecha: datetime = (getdate())
FK IdEstado: int
«PK»
+ PK_OrdenDeTrabajo(int)
«FK»
+ FK_OrdenDeTrabajo_Estado1(int)
+ FK_OrdenDeTrabajo_Usuario1(int)
(IdFormaDePago = IdFormaDePago)
«FK»
+PK_FormaDePago
FormaDePago
«column»
*PK IdFormaDePago: int
* Descripcion: nvarchar(50)
«PK»
+ PK_FormaDePago(int)

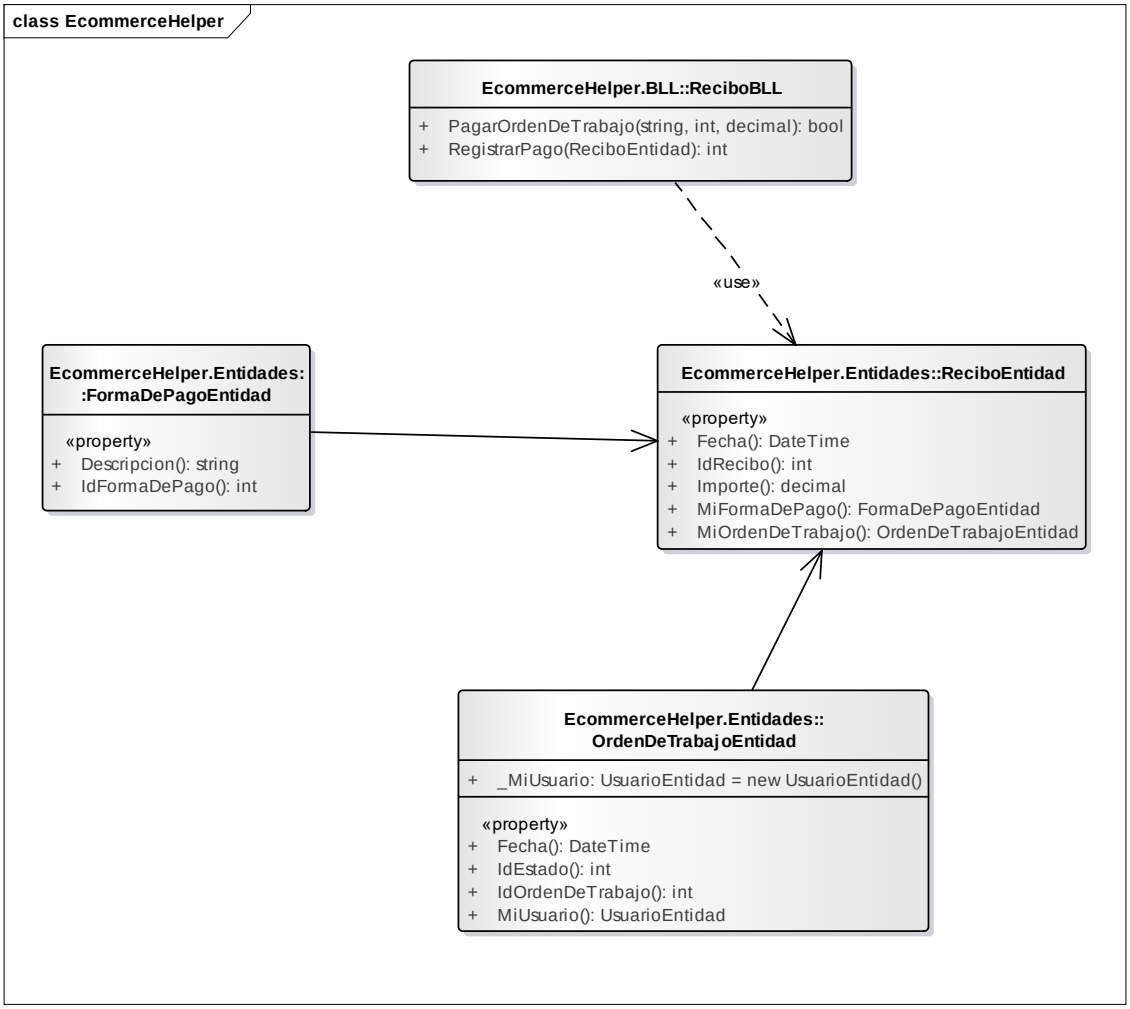
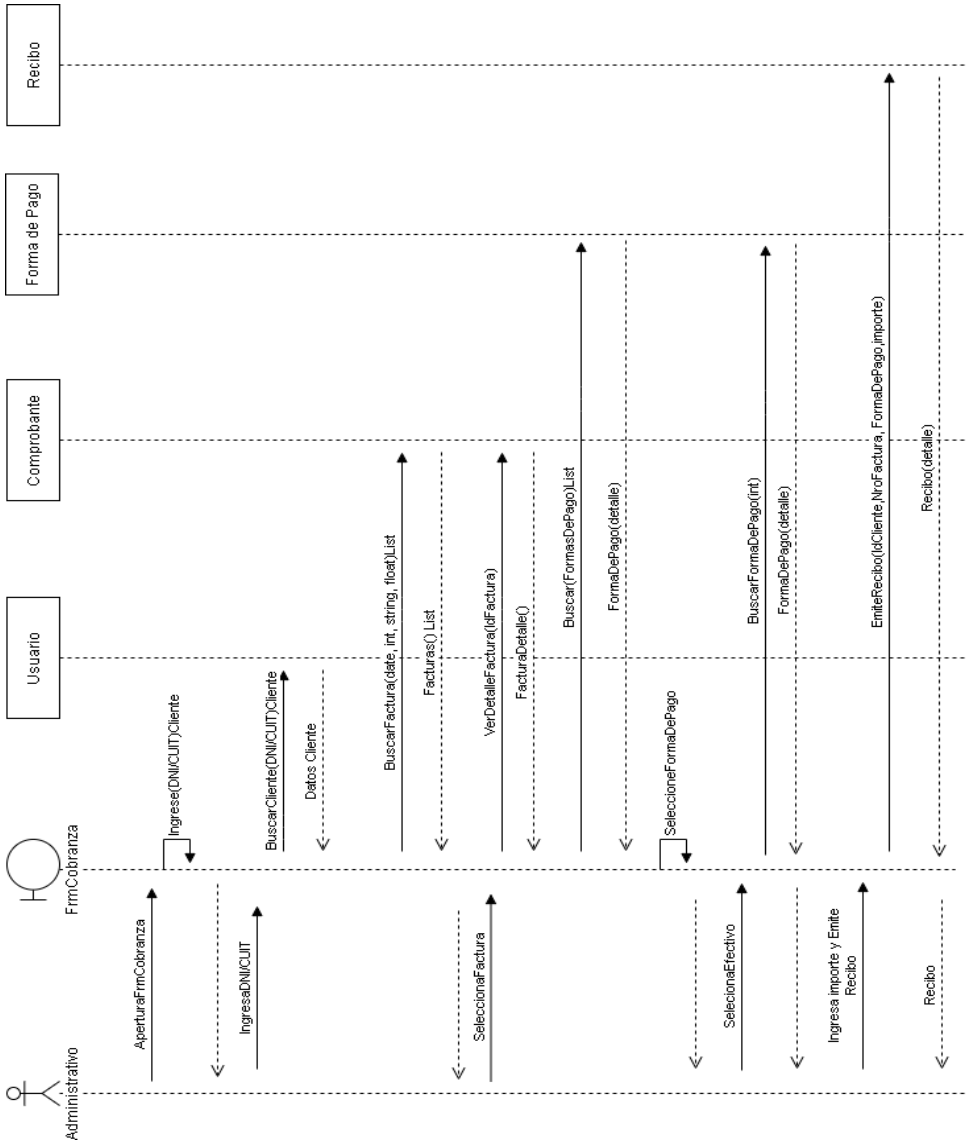

class EcommerceHelper
EcommerceHelper.BLL::ReciboBLL
+ PagarOrdenDeTrabajo(string, int, decimal): bool
+ RegistrarPago(ReciboEntidad): int
«use»
EcommerceHelper.Entidades::FormaDePagoEntidad
«property»
+ Descripcion(): string
+ IdFormaDePago(): int
EcommerceHelper.Entidades::ReciboEntidad
«property»
+ Fecha(): DateTime
+ IdRecibo(): int
+ Importe(): decimal
+ MiFormaDePago(): FormaDePagoEntidad
+ MiOrdenDeTrabajo(): OrdenDeTrabajoEntidad
EcommerceHelper.Entidades::OrdenDeTrabajoEntidad
+ _MiUsuario: UsuarioEntidad = new UsuarioEntidad()
«property»
+ Fecha(): DateTime
+ IdEstado(): int
+ IdOrdenDeTrabajo(): int
+ MiUsuario(): UsuarioEntidad

Diagrama de Secuencia CU - 044 -Genera Recibo



CU - 041 – Consultar Factura

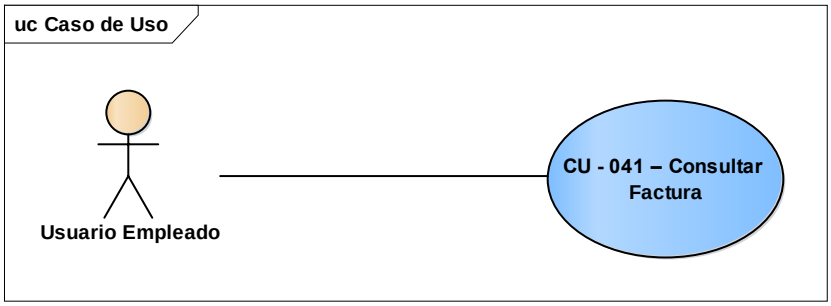
Precondición
Usuario empleado debe estar logueado.

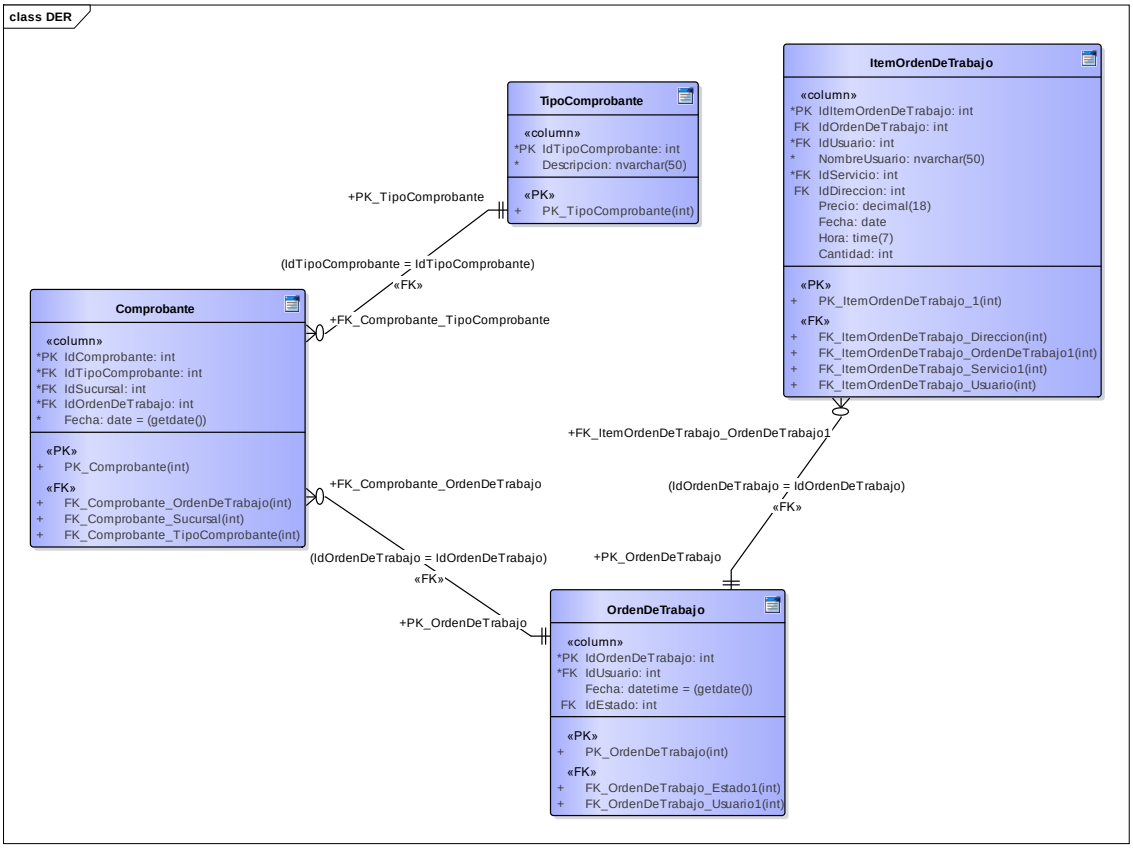
Actores primarios:
Cliente

Camino principal del caso de uso – Consulta Facturación

1. Usuario empleado ingresa a la sección de facturación.
2. El sistema lista la grilla con todas las facturas.
3. El usuario empleado selecciona la factura que desea consultar.
4. El sistema despliega el detalle de la factura seleccionada.

Postcondición
El usuario empleado visualizo el detalle de la factura seleccionada.


uc Caso de Uso
Usuario Empleado
CU - 041 – Consultar Factura


class DER
TipoComprobante
«column»
*PK IdTipoComprobante: int
* Descripcion: nvarchar(50)
«PK»
+ PK_TipoComprobante(int)
ItemOrdenDeTrabajo
«column»
*PK IdItemOrdenDeTrabajo: int
FK IdOrdenDeTrabajo: int
*FK IdUsuario: int
* NombreUsuario: nvarchar(50)
*FK IdServicio: int
FK IdDireccion: int
Precio: decimal(18)
Fecha: date
Hora: time(7)
Cantidad: int
«PK»
+ PK_ItemOrdenDeTrabajo_1(int)
«FK»
+ FK_ItemOrdenDeTrabajo_Direccion(int)
+ FK_ItemOrdenDeTrabajo_OrdenDeTrabajo1(int)
+ FK_ItemOrdenDeTrabajo_Servicio1(int)
+ FK_ItemOrdenDeTrabajo_Usuario(int)
+PK_TipoComprobante
(IdTipoComprobante = IdTipoComprobante)
«FK»
+FK_Comprobante_TipoComprobante
Comprobante
«column»
*PK IdComprobante: int
*FK IdTipoComprobante: int
*FK IdSucursal: int
*FK IdOrdenDeTrabajo: int
* Fecha: date = (getdate())
«PK»
+ PK_Comprobante(int)
«FK»
+ FK_Comprobante_OrdenDeTrabajo(int)
+ FK_Comprobante_Sucursal(int)
+ FK_Comprobante_TipoComprobante(int)
+FK_ItemOrdenDeTrabajo_OrdenDeTrabajo1
(IdOrdenDeTrabajo = IdOrdenDeTrabajo)
«FK»
+FK_Comprobante_OrdenDeTrabajo
(IdOrdenDeTrabajo = IdOrdenDeTrabajo)
«FK»
+PK_OrdenDeTrabajo
+PK_OrdenDeTrabajo
OrdenDeTrabajo
«column»
*PK IdOrdenDeTrabajo: int
*FK IdUsuario: int
Fecha: datetime = (getdate())
FK IdEstado: int
«PK»
+ PK_OrdenDeTrabajo(int)
«FK»
+ FK_OrdenDeTrabajo_Estado1(int)
+ FK_OrdenDeTrabajo_Usuario1(int)

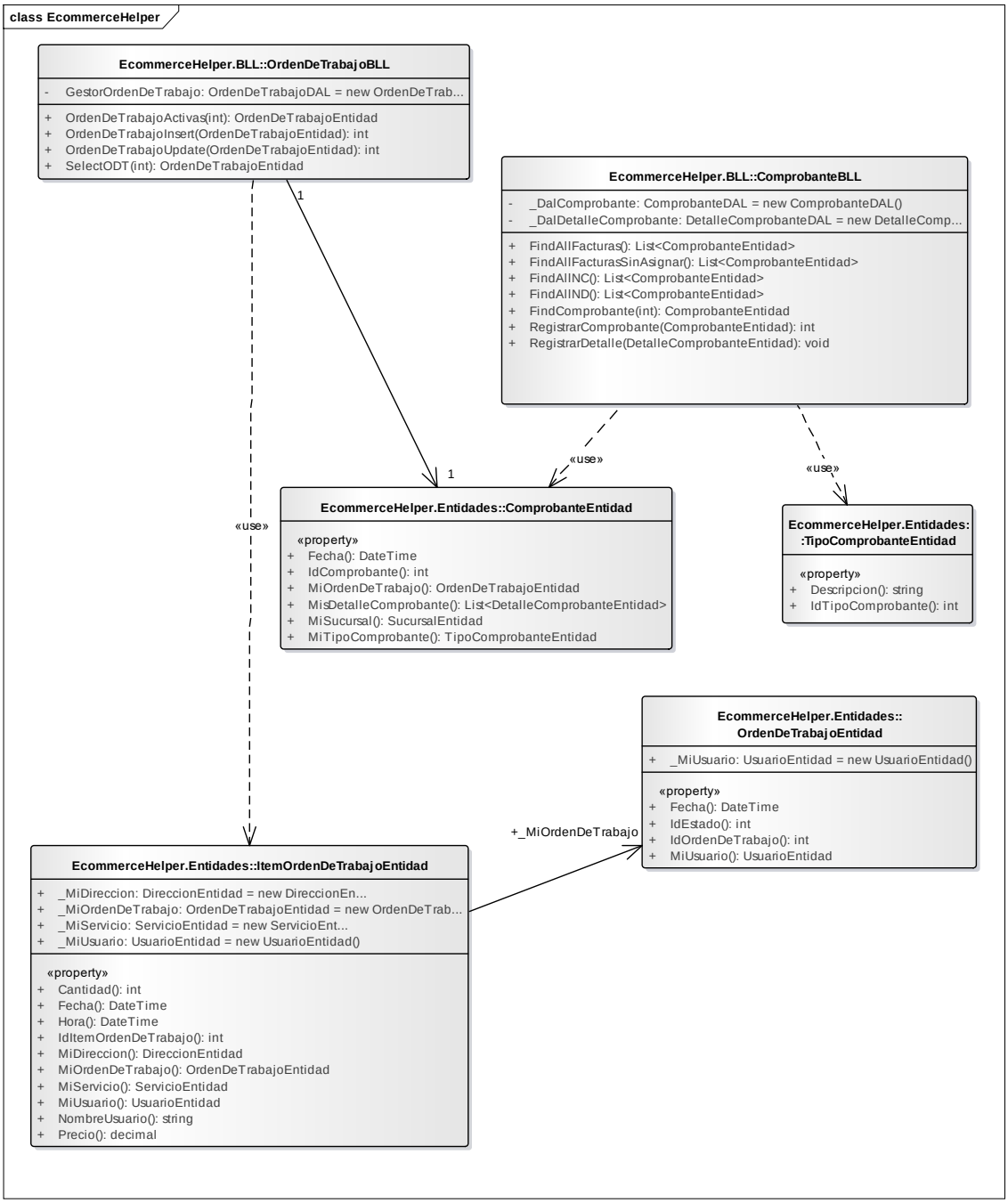
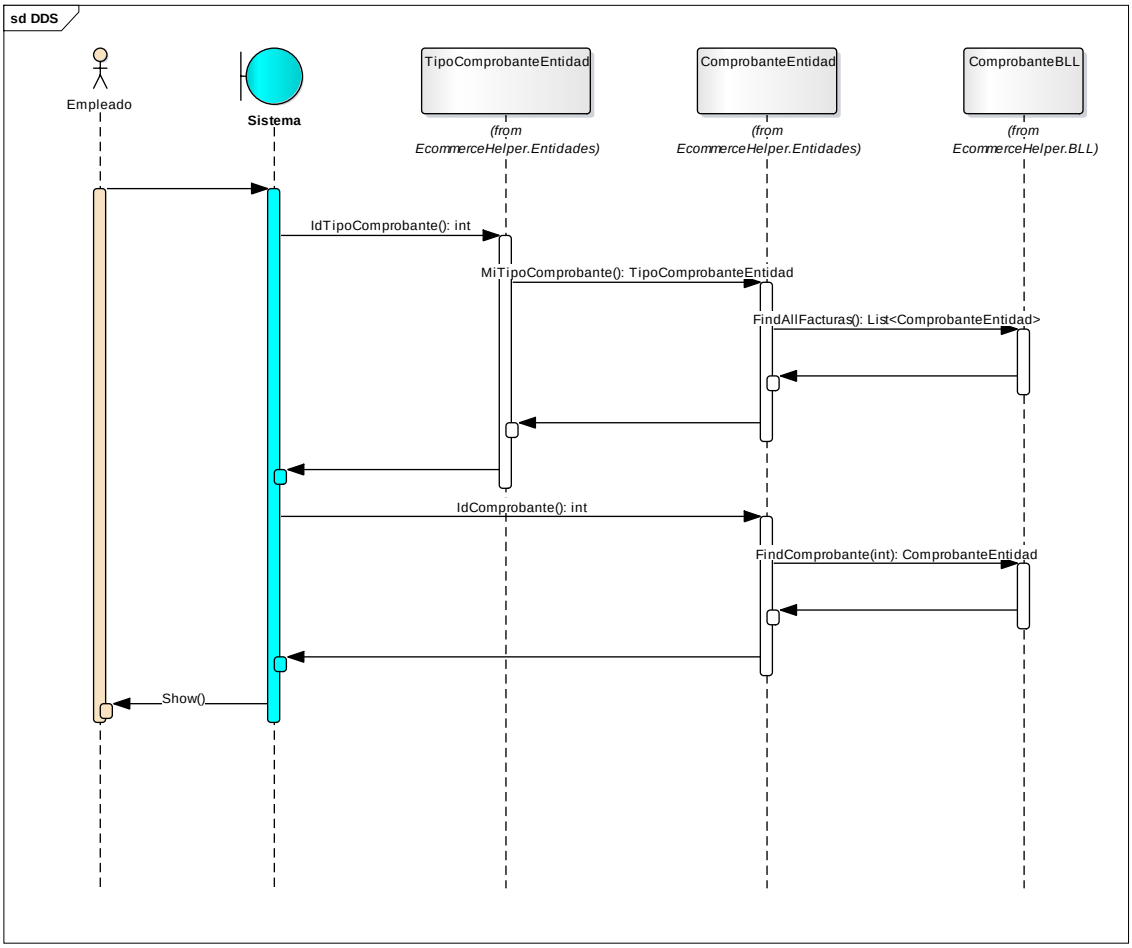
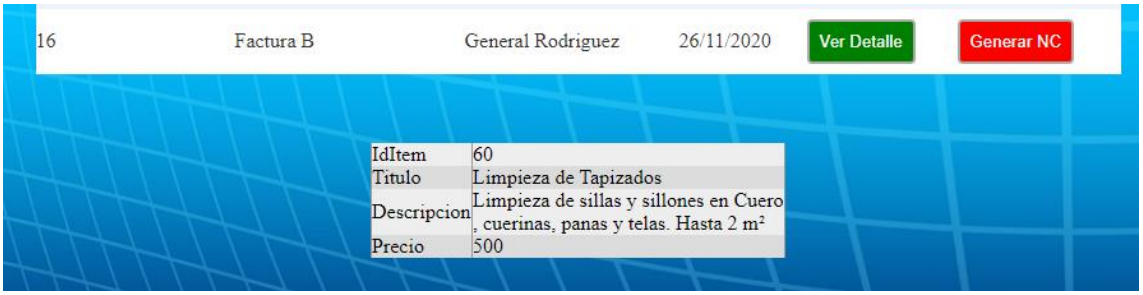

class EcommerceHelper
EcommerceHelper.BLL::OrdenDeTrabajoBLL
- GestorOrdenDeTrabajo: OrdenDeTrabajoDAL = new OrdenDeTrab...
+ OrdenDeTrabajoActivas(int): OrdenDeTrabajoEntidad
+ OrdenDeTrabajoInsert(OrdenDeTrabajoEntidad): int
+ OrdenDeTrabajoUpdate(OrdenDeTrabajoEntidad): int
+ SelectODT(int): OrdenDeTrabajoEntidad
EcommerceHelper.BLL::ComprobanteBLL
- _DalComprobante: ComprobanteDAL = new ComprobanteDAL()
- _DalDetalleComprobante: DetalleComprobanteDAL = new DetalleComp...
+ FindAllFacturas(): List<ComprobanteEntidad>
+ FindAllFacturasSinAsignar(): List<ComprobanteEntidad>
+ FindAllNC(): List<ComprobanteEntidad>
+ FindAllND(): List<ComprobanteEntidad>
+ FindComprobante(int): ComprobanteEntidad
+ RegistrarComprobante(ComprobanteEntidad): int
+ RegistrarDetalle(DetalleComprobanteEntidad): void
1
1
«use»
«use»
«use»
EcommerceHelper.Entidades::ComprobanteEntidad
«property»
+ Fecha(): DateTime
+ IdComprobante(): int
+ MiOrdenDeTrabajo(): OrdenDeTrabajoEntidad
+ MisDetalleComprobante(): List<DetalleComprobanteEntidad>
+ MiSucursal(): SucursalEntidad
+ MiTipoComprobante(): TipoComprobanteEntidad
EcommerceHelper.Entidades::TipoComprobanteEntidad
«property»
+ Descripcion(): string
+ IdTipoComprobante(): int
EcommerceHelper.Entidades::OrdenDeTrabajoEntidad
+ _MiUsuario: UsuarioEntidad = new UsuarioEntidad()
«property»
+ Fecha(): DateTime
+ IdEstado(): int
+ IdOrdenDeTrabajo(): int
+ MiUsuario(): UsuarioEntidad
+_MiOrdenDeTrabajo
EcommerceHelper.Entidades::ItemOrdenDeTrabajoEntidad
+ _MiDireccion: DireccionEntidad = new DireccionEn...
+ _MiOrdenDeTrabajo: OrdenDeTrabajoEntidad = new OrdenDeTrab...
+ _MiServicio: ServicioEntidad = new ServicioEnt...
+ _MiUsuario: UsuarioEntidad = new UsuarioEntidad()
«property»
+ Cantidad(): int
+ Fecha(): DateTime
+ Hora(): DateTime
+ IdItemOrdenDeTrabajo(): int
+ MiDireccion(): DireccionEntidad
+ MiOrdenDeTrabajo(): OrdenDeTrabajoEntidad
+ MiServicio(): ServicioEntidad
+ MiUsuario(): UsuarioEntidad
+ NombreUsuario(): string
+ Precio(): decimal

Diagrama de Secuencia CU - 041 – Consultar Factura



Pantalla


16
Factura B
General Rodriguez
26/11/2020
Ver Detalle
Generar NC
IdItem 60
Titulo Limpieza de Tapizados
Descripcion Limpieza de sillas y sillones en Cuero, cuerinas, panas y telas. Hasta 2 m²
Precio 500

Requisitos Especificaciones Adicionales

- CU - 017 - Multidioma
- CU - 018 – Registrar en Bitácora
- CU - 019 – Log In
- CU - 020 - Log Out
- CU - 022 – Modificar permisos a los Roles
- CU – 024 – Back Up
- CU – 025 – Restaurar Base de Datos
- CU – 026 – Ver Logs
- CU – 038 – Generar Digito Verificador Vertical
- CU – 39 – Generar Digito Verificador Horizontal
- CU – O45 – Consultar Roles

CU - 017 – Multidioma

Precondición:
El sitio por definición se despliega en idioma español.

Actores primarios:
Cliente ò usuarios de la página en general.

Camino principal del caso de uso – Multidioma

1. El sistema, en una lista desplegable le brinda una lista al usuario de los diferentes idiomas disponible.
2. El usuario selecciona el idioma deseado.
3. El usuario acepta.
4. Se despliega el detalle de la opción deseada.
5. El sistema abre la pantalla principal del sistema con el menú acorde al perfil e idioma del usuario

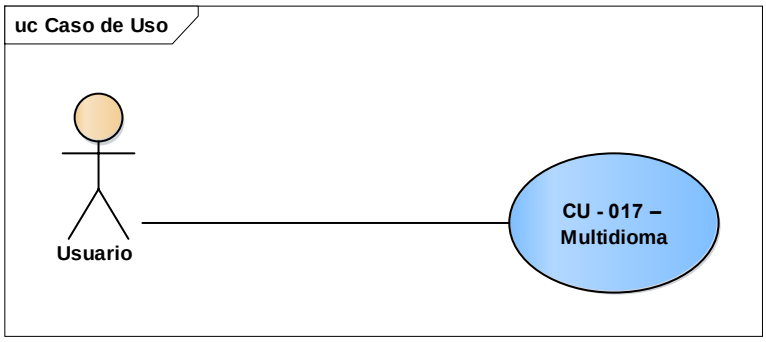
Caminos alternativos

Para el paso 3:

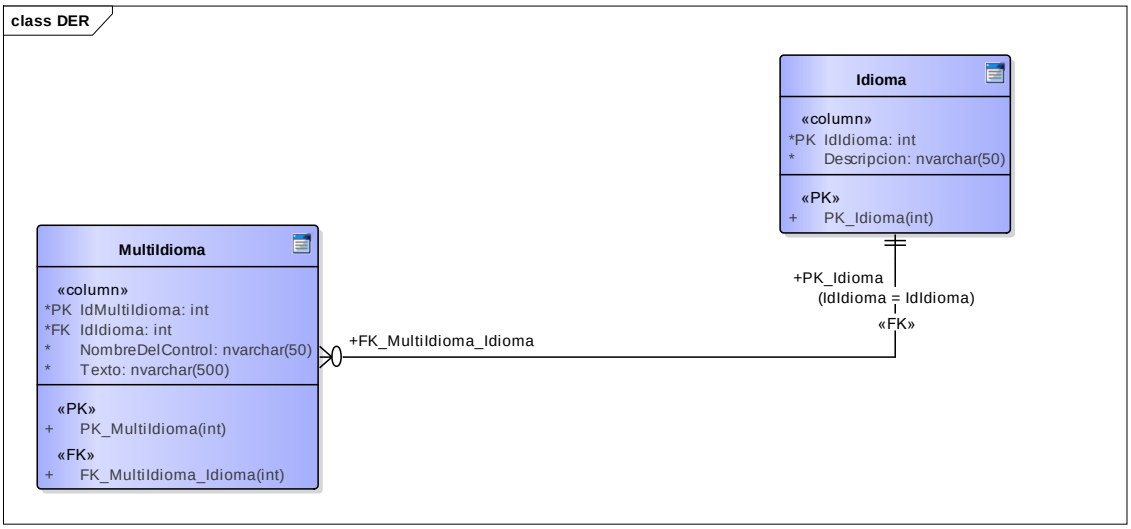
3a. El usuario cancela cambio de idioma.

3b. El sistema cancela el proceso de cambio de idioma retornando a la pantalla en la que el usuario se encontraba.

Postcondición:
El sitio se despliega en el idioma seleccionado o el definido automáticamente.


uc Caso de Uso
Usuario
CU - 017 – Multidioma

Español
Español
Ingles


class DER
Idioma
«column»
*PK IdIdioma: int
* Descripcion: nvarchar(50)
«PK»
+ PK_Idioma(int)
MultiIdioma
«column»
*PK IdMultiIdioma: int
*FK IdIdioma: int
* NombreDelControl: nvarchar(50)
* Texto: nvarchar(500)
«PK»
+ PK_MultiIdioma(int)
«FK»
+ FK_MultiIdioma_Idioma(int)
+FK_MultiIdioma_Idioma
+PK_Idioma
(IdIdioma = IdIdioma)
«FK»

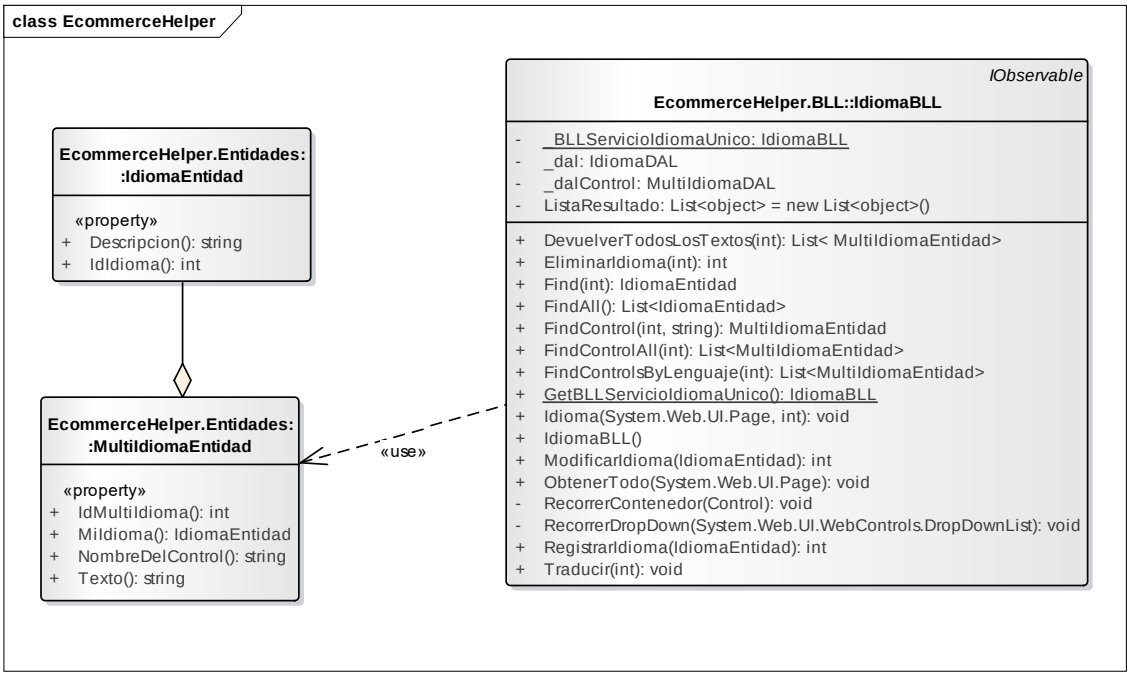
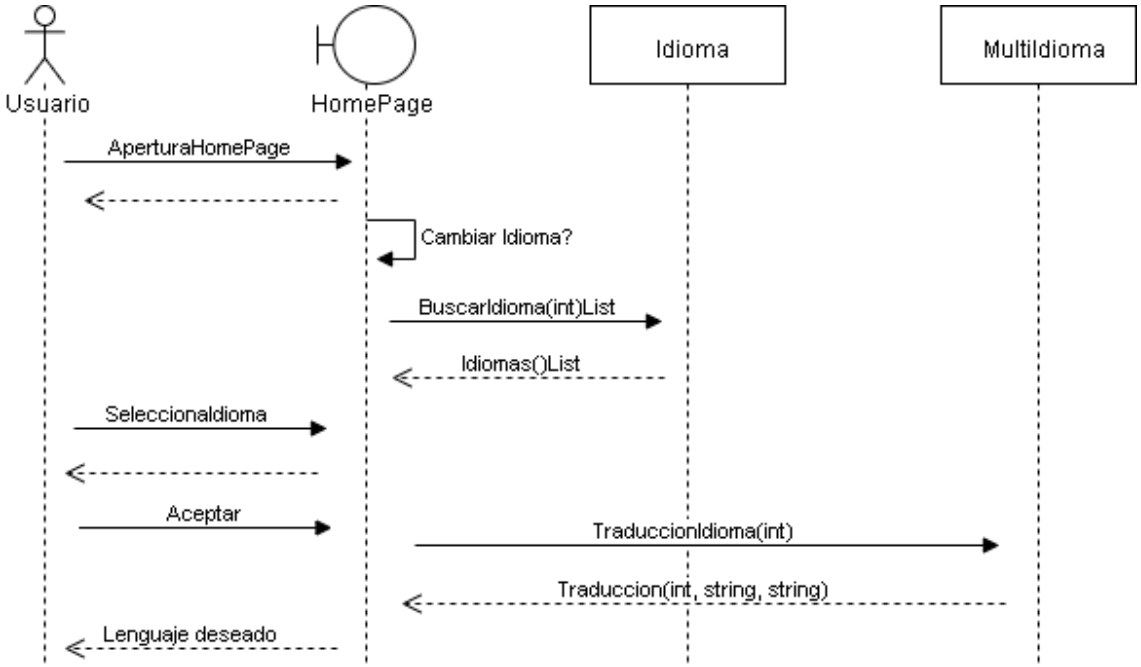

class EcommerceHelper
EcommerceHelper.Entidades::IdiomaEntidad
«property»
+ Descripcion(): string
+ IdIdioma(): int
EcommerceHelper.Entidades::MultiIdiomaEntidad
«property»
+ IdMultiIdioma(): int
+ MiIdioma(): IdiomaEntidad
+ NombreDelControl(): string
+ Texto(): string
«use»
IObservable
EcommerceHelper.BLL::IdiomaBLL
- _BLLServicioIdiomaUnico: IdiomaBLL
- _dal: IdiomaDAL
- _dalControl: MultiIdiomaDAL
- ListaResultado: List<object> = new List<object>()
+ DevuelverTodosLosTextos(int): List< MultiIdiomaEntidad>
+ EliminarIdioma(int): int
+ Find(int): IdiomaEntidad
+ FindAll(): List<IdiomaEntidad>
+ FindControl(int, string): MultiIdiomaEntidad
+ FindControlAll(int): List<MultiIdiomaEntidad>
+ FindControlsByLenguaje(int): List<MultiIdiomaEntidad>
+ GetBLLServicioIdiomaUnico(): IdiomaBLL
+ Idioma(System.Web.UI.Page, int): void
+ IdiomaBLL()
+ ModificarIdioma(IdiomaEntidad): int
+ ObtenerTodo(System.Web.UI.Page): void
- RecorrerContenedor(Control): void
- RecorrerDropDown(System.Web.UI.WebControls.DropDownList): void
+ RegistrarIdioma(IdiomaEntidad): int
+ Traducir(int): void

Diagrama de Secuencia CU - 017 – Multidioma



CU - 018 – Registrar en Bitácora

Precondición:
El usuario debe estar registrado en el sistema y debe realizar una acción que necesite ser guardada.

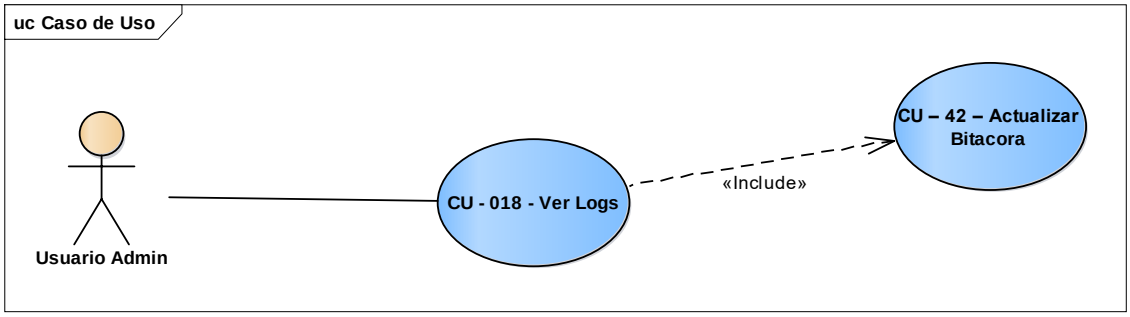
Actores primarios:
sistema.

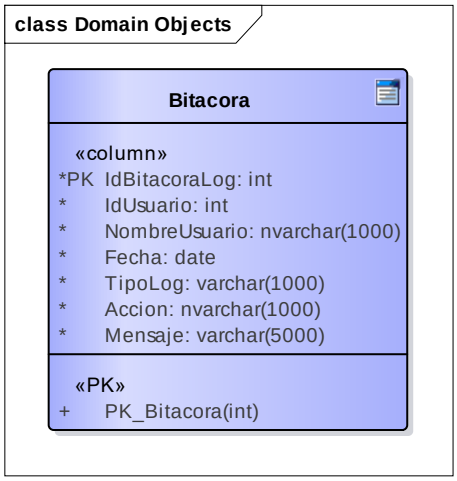
Camino principal del caso de uso – Registrar en Bitácora

1. El usuario realiza una acción dentro del sistema la cual debe quedar registrada.

1. El sistema guarda la fecha, hora, nombre de usuario y descripción de la acción realizada.

Postcondición:
Se guardan los datos necesarios para que quede registro de la acción realizada por el usuario.


uc Caso de Uso
Usuario Admin
CU - 018 - Ver Logs
«Include»
CU – 42 – Actualizar Bitacora


class Domain Objects
Bitacora
«column»
*PK IdBitacoraLog: int
* IdUsuario: int
* NombreUsuario: nvarchar(1000)
* Fecha: date
* TipoLog: varchar(1000)
* Accion: nvarchar(1000)
* Mensaje: varchar(5000)
«PK»
+ PK_Bitacora(int)

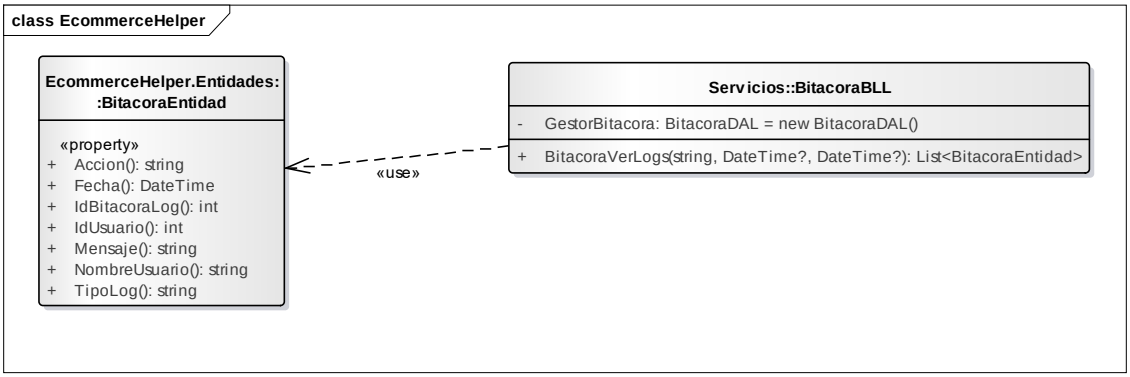
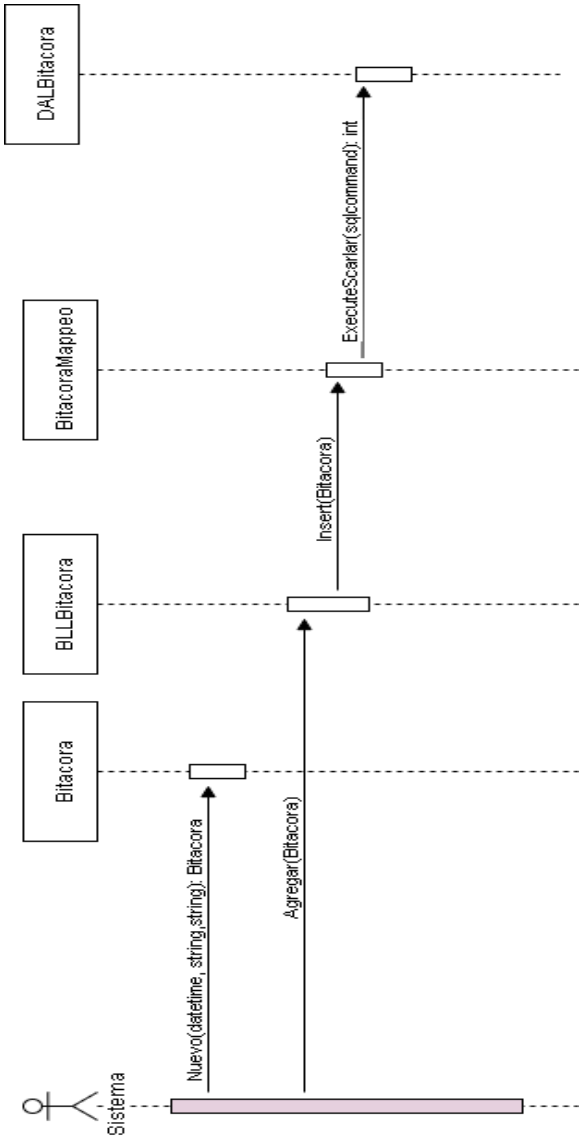

class EcommerceHelper
EcommerceHelper.Entidades::BitacoraEntidad
«property»
+ Accion(): string
+ Fecha(): DateTime
+ IdBitacoraLog(): int
+ IdUsuario(): int
+ Mensaje(): string
+ NombreUsuario(): string
+ TipoLog(): string
«use»
Servicios::BitacoraBLL
- GestorBitacora: BitacoraDAL = new BitacoraDAL()
+ BitacoraVerLogs(string, DateTime?, DateTime?): List<BitacoraEntidad>

Diagrama de Secuencia CU - 018 – Ver Logs



CU - 019 – Log in

Precondición:
El usuario debe estar registrado en el sistema.

Actores primarios:
Cliente ò usuarios de la página en general.

Camino principal del caso de uso – Log In

1. El usuario inicia el sistema.
2. El usuario ingresa Usuario y contraseña.
3. El sistema realiza la encriptación del usuario y de la clave del usuario.
4. El sistema controla que el id usuario y contraseña ingresados encuentre registrado.
5. El sistema carga los permisos correspondientes definidos en la creación del usuario.
6. El sistema graba en la bitácora la información del inicio de sesión.
7. El sistema abre la pantalla principal del sistema con el menú acorde al perfil del usuario.

Caminos alternativos

Para el paso 2:

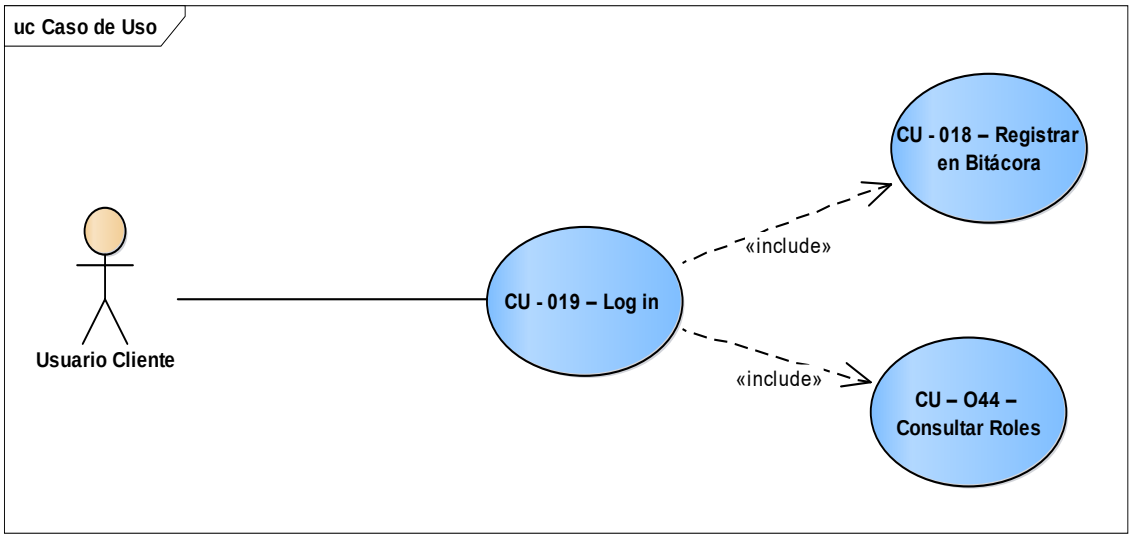
2a. El usuario cancela cambio iniciar sesión.

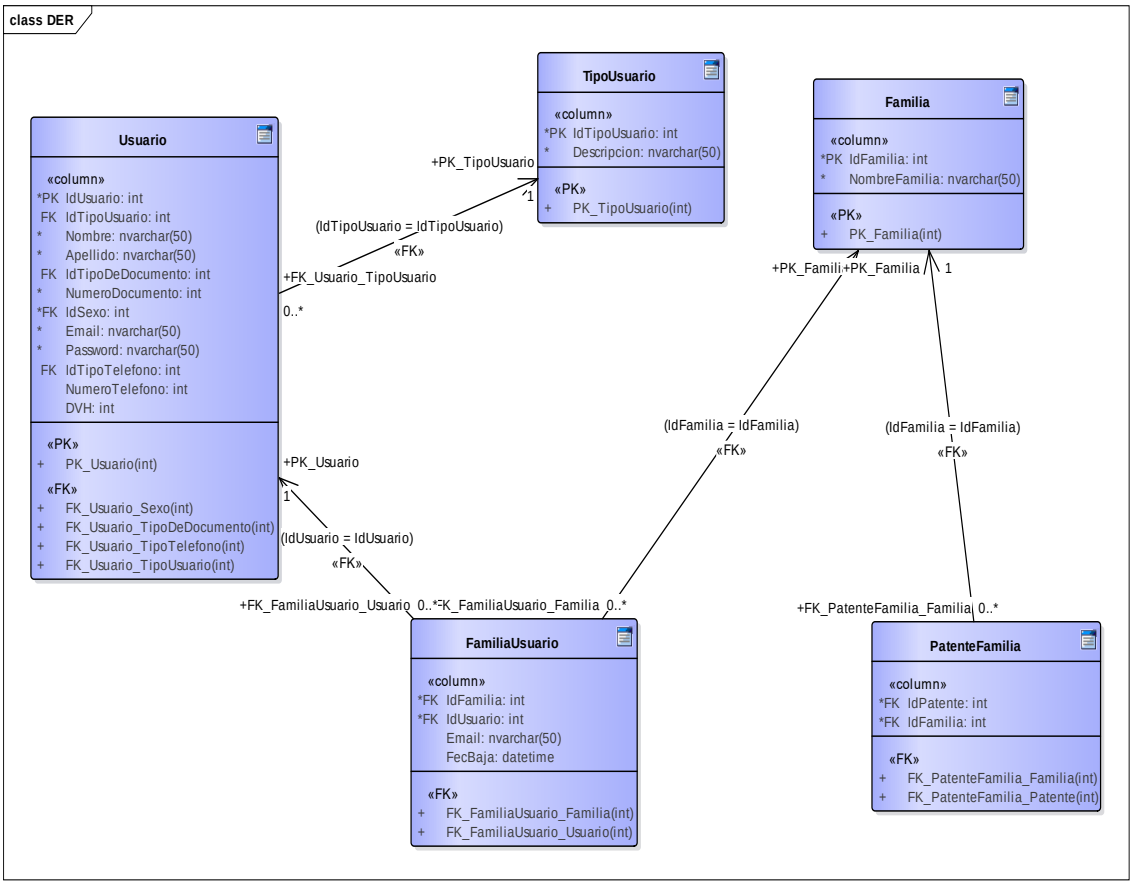
Para el paso 4:

4a. El sistema valida que la combinación de usuario y contraseña no existe. Si el usuario no existe, incrementa el contador de accesos incorrectos por usuario. Luego, informa a través de un mensaje de error al usuario. Volver al punto 1.

4b. Luego de 3 accesos fallidos la cuenta es bloqueada. El sistema bloquea la cuenta de usuario. Se graba en bitácora el tercer intento fallido de inicio de sesión.

Postcondición:
El usuario tiene una sesión iniciada.


uc Caso de Uso
Usuario Cliente
CU - 019 – Log in
«include»
CU - 018 – Registrar en Bitácora
«include»
CU – O44 – Consultar Roles


class DER
TipoUsuario
«column»
*PK IdTipoUsuario: int
* Descripcion: nvarchar(50)
«PK»
+ PK_TipoUsuario(int)
Familia
«column»
*PK IdFamilia: int
* NombreFamilia: nvarchar(50)
«PK»
+ PK_Familia(int)
Usuario
«column»
*PK IdUsuario: int
FK IdTipoUsuario: int
* Nombre: nvarchar(50)
* Apellido: nvarchar(50)
FK IdTipoDeDocumento: int
* NumeroDocumento: int
*FK IdSexo: int
* Email: nvarchar(50)
* Password: nvarchar(50)
FK IdTipoTelefono: int
NumeroTelefono: int
DVH: int
«PK»
+ PK_Usuario(int)
«FK»
+ FK_Usuario_Sexo(int)
+ FK_Usuario_TipoDeDocumento(int)
+ FK_Usuario_TipoTelefono(int)
+ FK_Usuario_TipoUsuario(int)
+PK_TipoUsuario
1
(IdTipoUsuario = IdTipoUsuario)
«FK»
+FK_Usuario_TipoUsuario
0..*
+PK_Familia
1
(IdFamilia = IdFamilia)
«FK»
(IdFamilia = IdFamilia)
«FK»
+PK_Usuario
1
(IdUsuario = IdUsuario)
«FK»
+FK_FamiliaUsuario_Usuario 0..*
+FK_FamiliaUsuario_Familia 0..*
+FK_PatenteFamilia_Familia 0..*
FamiliaUsuario
«column»
*FK IdFamilia: int
*FK IdUsuario: int
Email: nvarchar(50)
FecBaja: datetime
«FK»
+ FK_FamiliaUsuario_Familia(int)
+ FK_FamiliaUsuario_Usuario(int)
PatenteFamilia
«column»
*FK IdPatente: int
*FK IdFamilia: int
«FK»
+ FK_PatenteFamilia_Familia(int)
+ FK_PatenteFamilia_Patente(int)

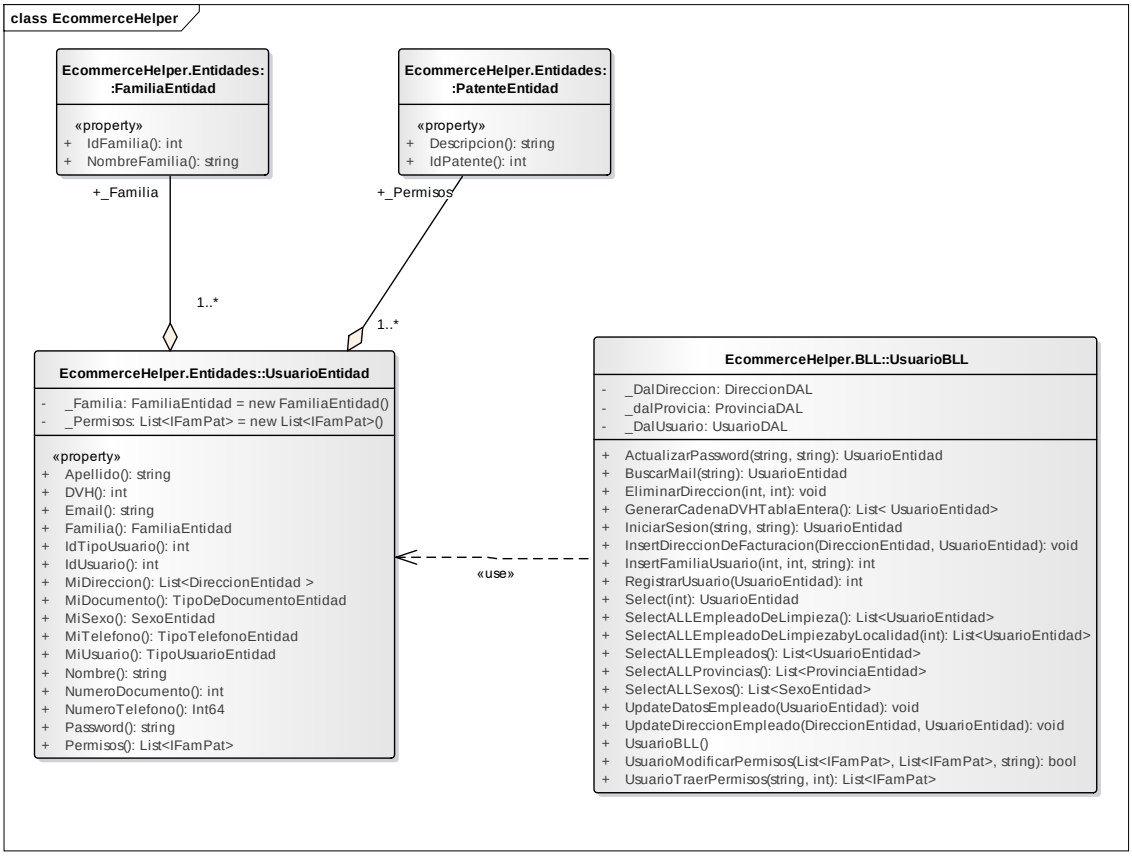
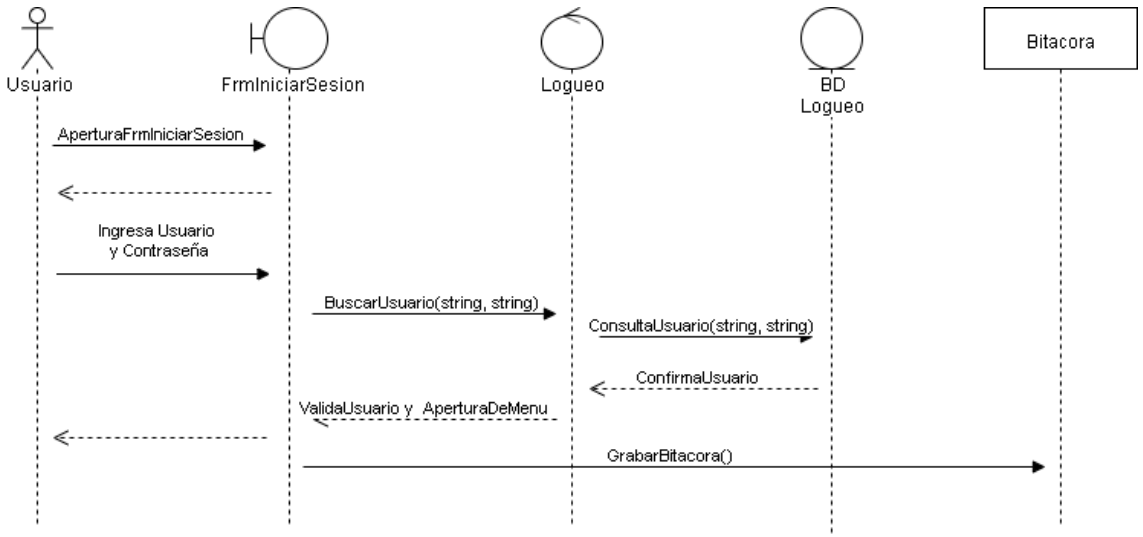
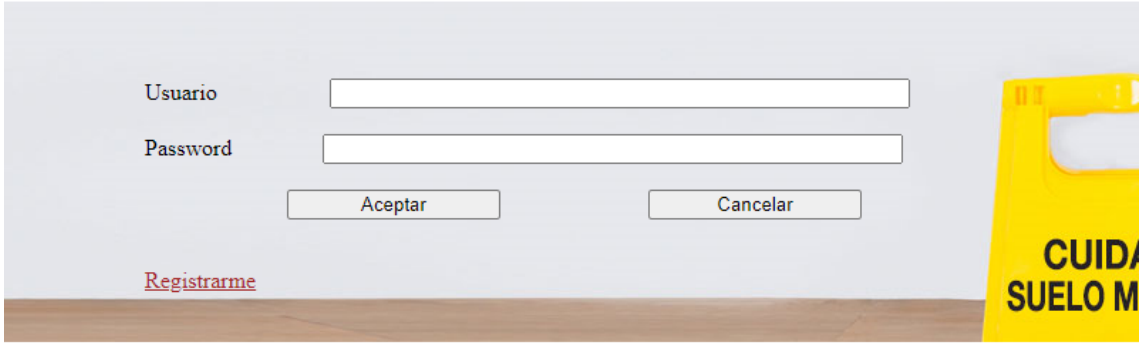

class EcommerceHelper
EcommerceHelper.Entidades::FamiliaEntidad
«property»
+ IdFamilia(): int
+ NombreFamilia(): string
EcommerceHelper.Entidades::PatenteEntidad
«property»
+ Descripcion(): string
+ IdPatente(): int
+_Familia
+_Permisos
1..*
1..*
EcommerceHelper.Entidades::UsuarioEntidad
- _Familia: FamiliaEntidad = new FamiliaEntidad()
- _Permisos: List<IFamPat> = new List<IFamPat>()
«property»
+ Apellido(): string
+ DVH(): int
+ Email(): string
+ Familia(): FamiliaEntidad
+ IdTipoUsuario(): int
+ IdUsuario(): int
+ MiDireccion(): List<DireccionEntidad >
+ MiDocumento(): TipoDeDocumentoEntidad
+ MiSexo(): SexoEntidad
+ MiTelefono(): TipoTelefonoEntidad
+ MiUsuario(): TipoUsuarioEntidad
+ Nombre(): string
+ NumeroDocumento(): int
+ NumeroTelefono(): Int64
+ Password(): string
+ Permisos(): List<IFamPat>
«use»
EcommerceHelper.BLL::UsuarioBLL
- _DalDireccion: DireccionDAL
- _dalProvicia: ProvinciaDAL
- _DalUsuario: UsuarioDAL
+ ActualizarPassword(string, string): UsuarioEntidad
+ BuscarMail(string): UsuarioEntidad
+ EliminarDireccion(int, int): void
+ GenerarCadenaDVHTablaEntera(): List< UsuarioEntidad>
+ IniciarSesion(string, string): UsuarioEntidad
+ InsertDireccionDeFacturacion(DireccionEntidad, UsuarioEntidad): void
+ InsertFamiliaUsuario(int, int, string): int
+ RegistrarUsuario(UsuarioEntidad): int
+ Select(int): UsuarioEntidad
+ SelectALLEmpleadoDeLimpieza(): List<UsuarioEntidad>
+ SelectALLEmpleadoDeLimpiezabyLocalidad(int): List<UsuarioEntidad>
+ SelectALLEmpleados(): List<UsuarioEntidad>
+ SelectALLProvincias(): List<ProvinciaEntidad>
+ SelectALLSexos(): List<SexoEntidad>
+ UpdateDatosEmpleado(UsuarioEntidad): void
+ UpdateDireccionEmpleado(DireccionEntidad, UsuarioEntidad): void
+ UsuarioBLL()
+ UsuarioModificarPermisos(List<IFamPat>, List<IFamPat>, string): bool
+ UsuarioTraerPermisos(string, int): List<IFamPat>

Diagrama de Secuencia CU - 019 – Log in



Pantalla

Iniciar Sesion



CopyRight Argentina 2020 - Helper

CU - 020 – Log Out

Precondición:
El usuario debe tener la sesión iniciada en el sistema.

Actores primarios:
Cliente ò usuarios de la página en general.

Camino principal del caso de uso – Log Out

1. El usuario selecciona la opción de cierre de sesión.
2. El sistema consulta la confirmación del cierre de sesión.
3. El usuario selecciona la confirmación de cerrar la sesión.
4. El sistema graba en la bitácora la información del inicio de sesión.

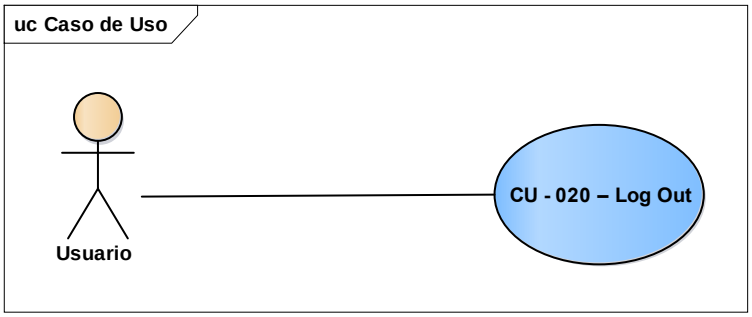
Caminos alternativos

Para el paso 3:

3a. El usuario selecciona que no confirma el cierre de la sesión.

3b. El sistema cancela el cierre de sesión.

Postcondición:
El usuario tiene una sesión cerrada.


uc Caso de Uso
Usuario
CU - 020 – Log Out

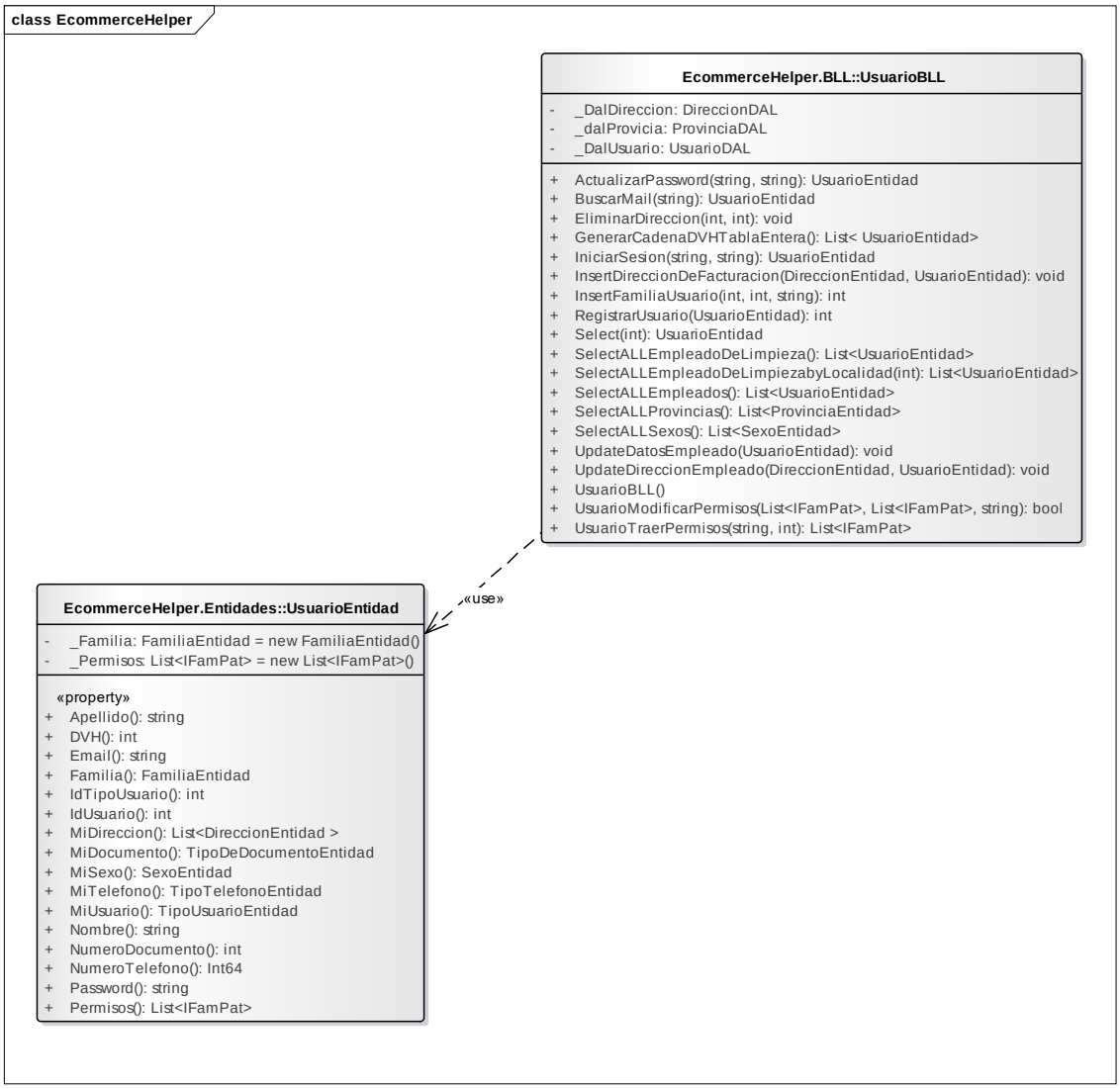
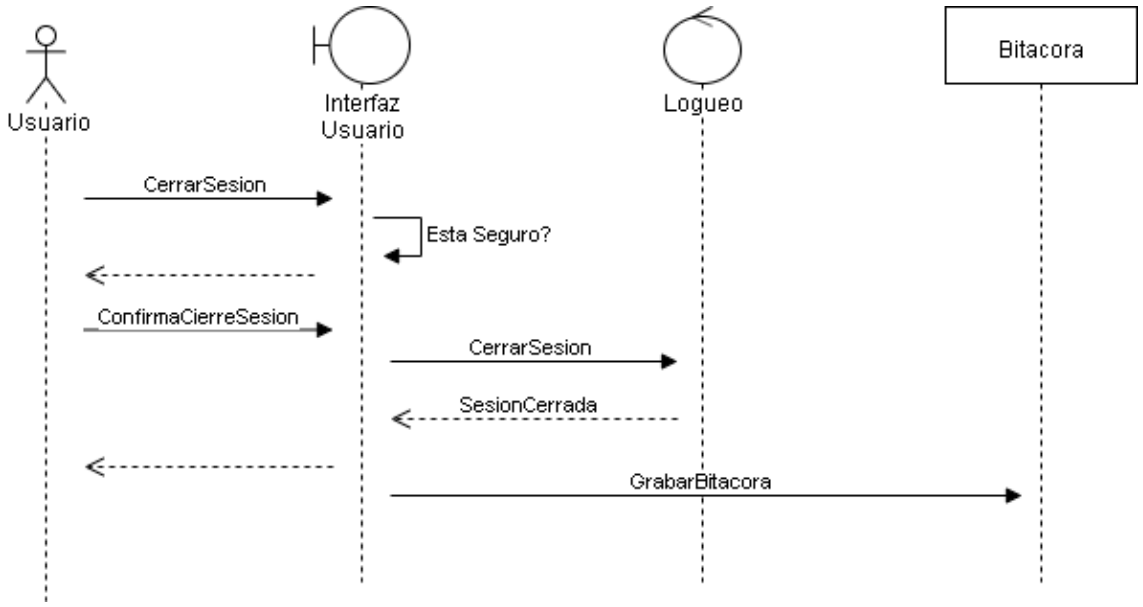

class EcommerceHelper
EcommerceHelper.BLL::UsuarioBLL
- _DalDireccion: DireccionDAL
- _dalProvicia: ProvinciaDAL
- _DalUsuario: UsuarioDAL
+ ActualizarPassword(string, string): UsuarioEntidad
+ BuscarMail(string): UsuarioEntidad
+ EliminarDireccion(int, int): void
+ GenerarCadenaDVHTablaEntera(): List< UsuarioEntidad>
+ IniciarSesion(string, string): UsuarioEntidad
+ InsertDireccionDeFacturacion(DireccionEntidad, UsuarioEntidad): void
+ InsertFamiliaUsuario(int, int, string): int
+ RegistrarUsuario(UsuarioEntidad): int
+ Select(int): UsuarioEntidad
+ SelectALLEmpleadoDeLimpieza(): List<UsuarioEntidad>
+ SelectALLEmpleadoDeLimpiezabyLocalidad(int): List<UsuarioEntidad>
+ SelectALLEmpleados(): List<UsuarioEntidad>
+ SelectALLProvincias(): List<ProvinciaEntidad>
+ SelectALLSexos(): List<SexoEntidad>
+ UpdateDatosEmpleado(UsuarioEntidad): void
+ UpdateDireccionEmpleado(DireccionEntidad, UsuarioEntidad): void
+ UsuarioBLL()
+ UsuarioModificarPermisos(List<IFamPat>, List<IFamPat>, string): bool
+ UsuarioTraerPermisos(string, int): List<IFamPat>
«use»
EcommerceHelper.Entidades::UsuarioEntidad
- _Familia: FamiliaEntidad = new FamiliaEntidad()
- _Permisos: List<IFamPat> = new List<IFamPat>()
«property»
+ Apellido(): string
+ DVH(): int
+ Email(): string
+ Familia(): FamiliaEntidad
+ IdTipoUsuario(): int
+ IdUsuario(): int
+ MiDireccion(): List<DireccionEntidad >
+ MiDocumento(): TipoDeDocumentoEntidad
+ MiSexo(): SexoEntidad
+ MiTelefono(): TipoTelefonoEntidad
+ MiUsuario(): TipoUsuarioEntidad
+ Nombre(): string
+ NumeroDocumento(): int
+ NumeroTelefono(): Int64
+ Password(): string
+ Permisos(): List<IFamPat>

Diagrama de Secuencia CU - 020 – Log Out



CU – 024 - Back Up

Pre-condición:
El Administrador debe estar logueado

Actores primarios:
Administrador

Camino principal de CU – 024 - Back Up: Realizar backup de base de datos

1. El Administrador se dirige a la sección “Base de Datos – Backup”
2. El Sistema redirige a la sección “Backup” y presenta el campo a completar para efectuar el resguardo de la base de datos:
    a. Nombre del archivo: Texto
    b. Opción “Backup” para realizar el resguardo de la base de datos
3. El Administrador completa el Nombre requerido y selecciona la opción “Backup”
4. El Sistema verifica que el Nombre del archivo se haya ingresado
5. El Sistema emite un mensaje de confirmación: “¿Está Seguro?”
6. El Administrador confirma la operación mediante la opción “Aceptar”
7. El Sistema realiza el backup de la base de datos y le envía al navegador web el archivo .bak a ser descargado en la PC del Administrador

Caminos Alternativos

Para el paso 4: El Sistema comprueba que el Nombre del archivo no fue ingresado

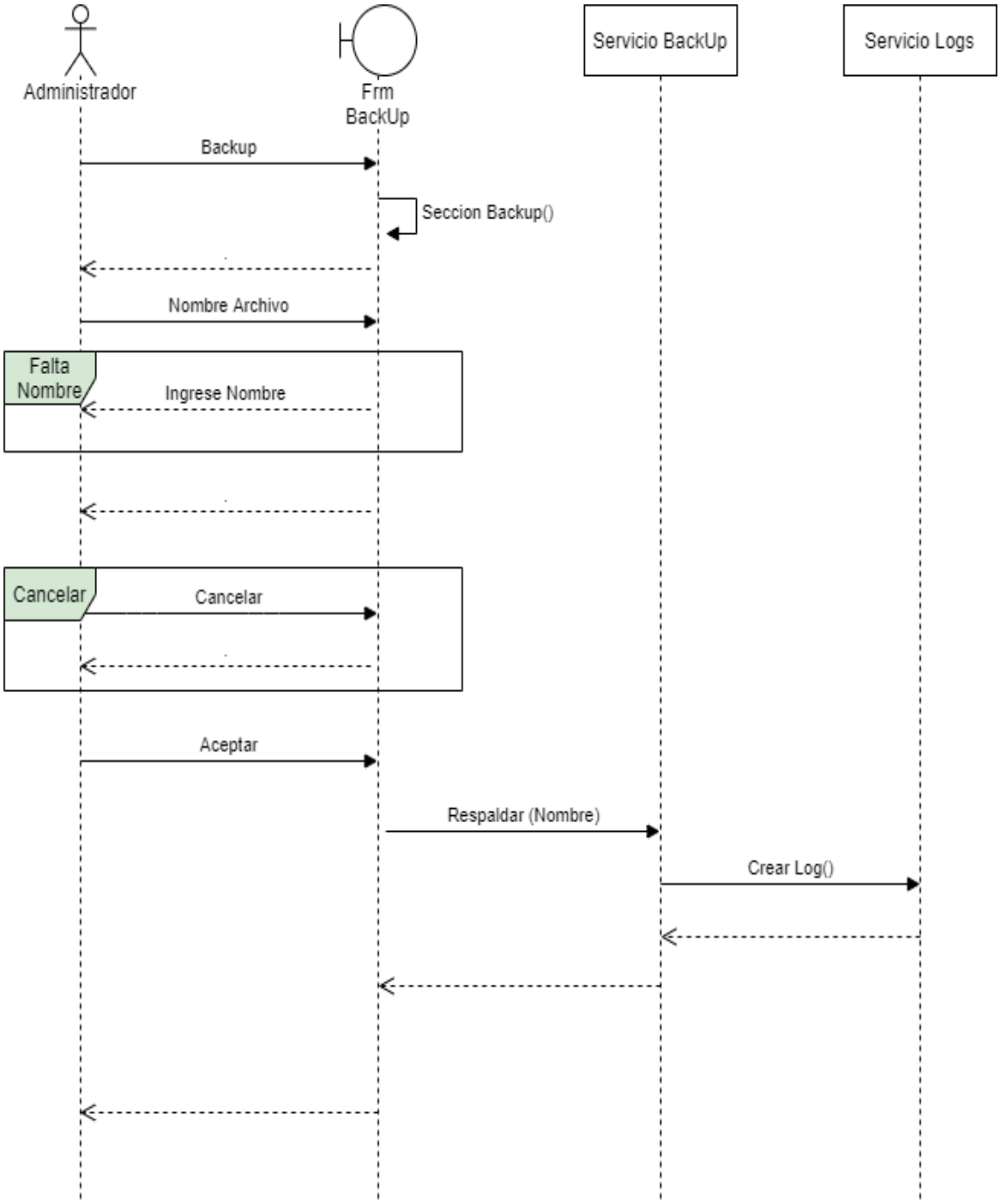
4.1 El Sistema presenta en pantalla un mensaje indicando que se ingrese el nombre del archivo
4.2 El curso continúa en el paso 3 del camino principal

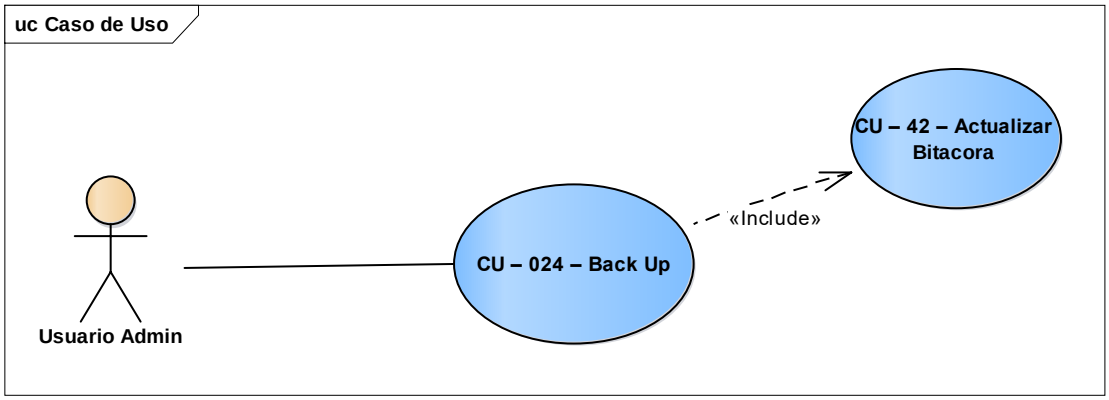
Para el paso 5: El Administrador selecciona la opción “Cancelar”

5.1 El Sistema cierra el mensaje de confirmación

Post-condición:
Se genera un archivo .bak con un restore de la base de datos

Diagrama de Secuencia CU - 024 – BackUp




uc Caso de Uso
CU – 42 – Actualizar Bitacora
«Include»
CU – 024 – Back Up
Usuario Admin

class EcommerceHelper
Seguridad::ServicioBackup
+ Respaldar(string): bool
+ Restaurar(string, string): bool

Pantalla

CU – 025- Restore

Pre-condición:
El Administrador debe estar logueado

Actores primarios:

Administrador

Camino principal de CU- 025 - Restore: Restaurar base de datos

1. El Administrador se dirige a la sección “Base de Datos - Restaurar”
2. El Sistema redirige a la sección “Restaurar” y presenta los campos a completar para efectuar la restauración de la base de datos:
    a. Nombre de la base de datos: Texto
    b. Opción “Seleccionar archivo” para seleccionar la ubicación del archivo .bak
3. El Administrador completa el campo Nombre, selecciona la ubicación del archivo .bak y selecciona la opción “Restaurar”
4. El Sistema verifica que el Nombre de la base de datos y la ruta del archivo .bak se hayan ingresado
5. El Sistema emite un mensaje de confirmación: “¿Está Seguro?”
6. El Administrador confirma la operación mediante la opción “Aceptar”
7. El Sistema realiza la restauración de la base de datos desde el archivo con extensión .bak de la ruta proporcionada, cierra la sesión del usuario y redirige a la pantalla principal

Caminos Alternativos
Para el paso 4: El Sistema comprueba que el Nombre de la base de datos no fue ingresado

4.1 El Sistema presenta en pantalla un mensaje indicando que se ingrese el nombre de la base de datos

Para el paso 4’: El Sistema comprueba que el archivo .bak no fue ingresado

4’.1 El Sistema presenta en pantalla un mensaje indicando que se seleccione el archivo .bak
4’.2 El curso continúa en el paso 3 del camino principal

Para el paso 4'': El Sistema comprueba que el archivo seleccionado no tiene formato .bak

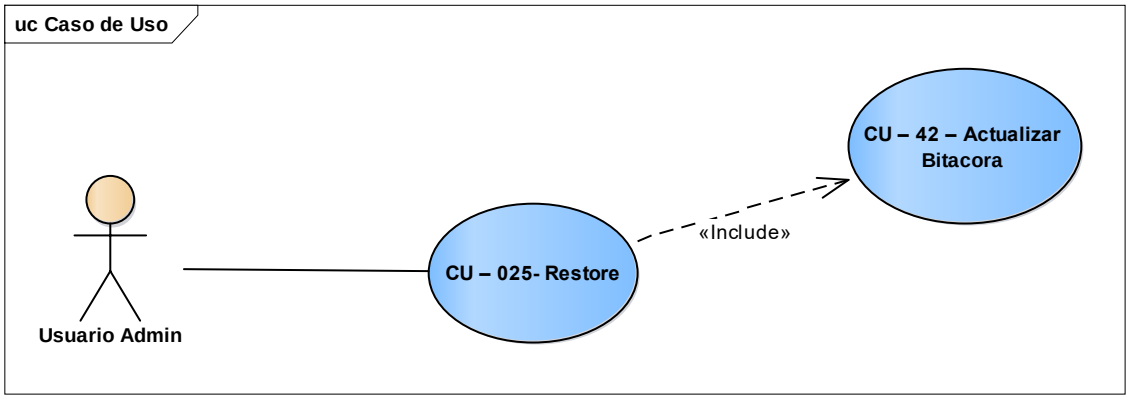
4''.1 El Sistema presenta en pantalla un mensaje indicando que únicamente se permiten archivos .bak
4''.2 El curso continúa en el paso 3 del camino principal

Para el paso 5: El Administrador selecciona la opción "Cancelar"

5.1 El Sistema cierra el mensaje de confirmación

Post-condición:

Se restaura la base de datos al estado provisto por el archivo .bak

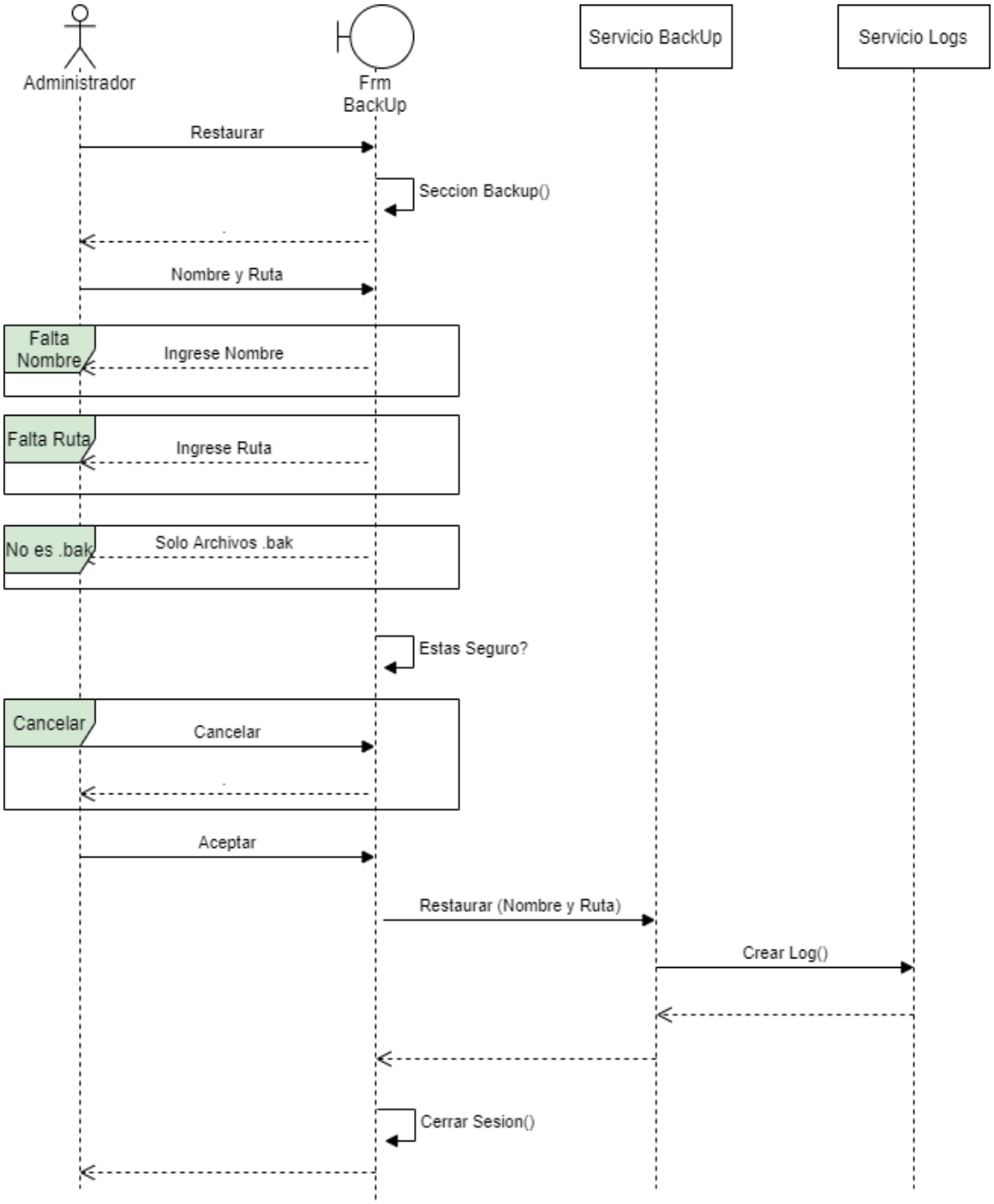

uc Caso de Uso
CU – 42 – Actualizar Bitacora
«Include»
CU – 025- Restore
Usuario Admin

class EcommerceHelper
Seguridad::ServicioBackup
+ Respaldar(string): bool
+ Restaurar(string, string): bool

Pantalla

Restaurar
Nombre
Nombre Base de Datos
Seleccionar archivo
Ningún archivo seleccionado
Restaurar
Cancelar

Diagrama de Secuencia CU - 025 – Restore



CU - 026 – Ver Logs

Precondición:
El administrador debe estar logueado.

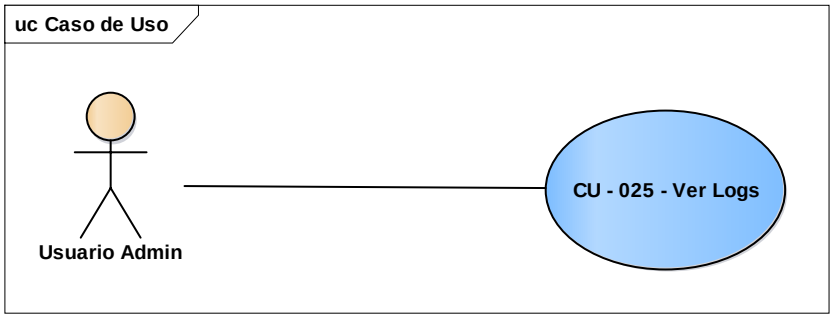
Actores primarios:
Administrador.

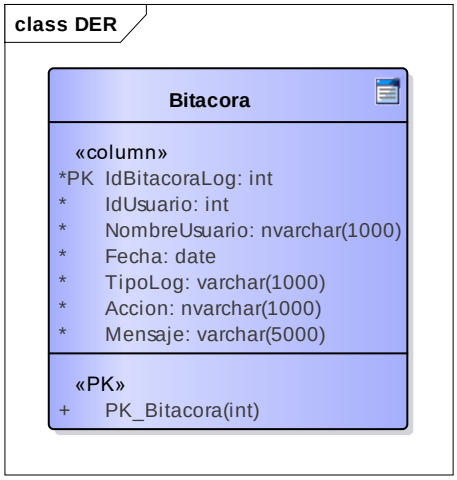
Camino principal del caso de uso – Ver Logs

1. El Administrador se dirige a la sección de la Intranet “Menú Administración - Bitácora”
2. El Sistema redirige a la pantalla “Ver Logs” y presenta las opciones de filtrado de eventos:
    a. Fecha Inicio
    b. Fecha Fin
    c. Tipo Log
3. El Administrador realiza el filtro que requiera y selecciona la opción “Buscar”
4. El Sistema obtiene los logs según el filtro aplicado por el Administrador y los muestra en pantalla:
    a. Número de evento
    b. Nombre de Usuario
    c. Fecha
    d. Tipo de evento
    e. Acción
    f. Mensaje

Caminos Alternativos
No existen caminos alternativos

Postcondición:
Se guardan los datos necesarios para que quede registro de la acción realizada por el usuario.


uc Caso de Uso
Usuario Admin
CU - 025 - Ver Logs


class DER
Bitacora
«column»
*PK IdBitacoraLog: int
* IdUsuario: int
* NombreUsuario: nvarchar(1000)
* Fecha: date
* TipoLog: varchar(1000)
* Accion: nvarchar(1000)
* Mensaje: varchar(5000)
«PK»
+ PK_Bitacora(int)

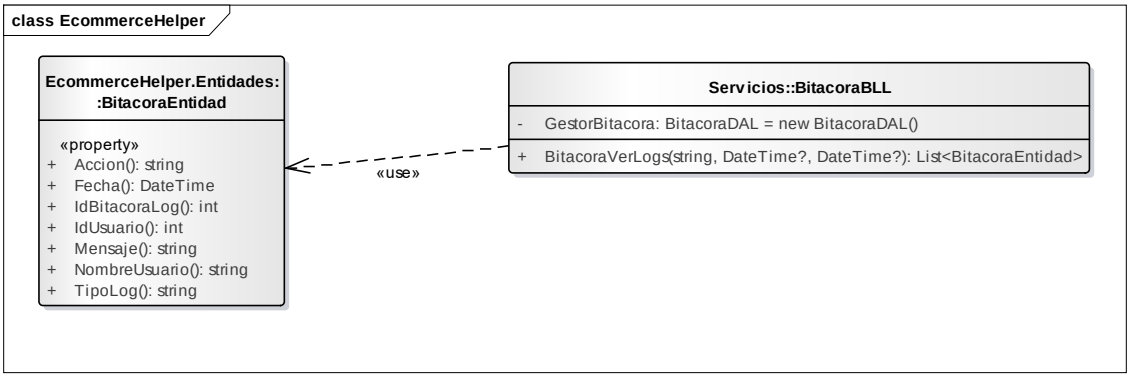
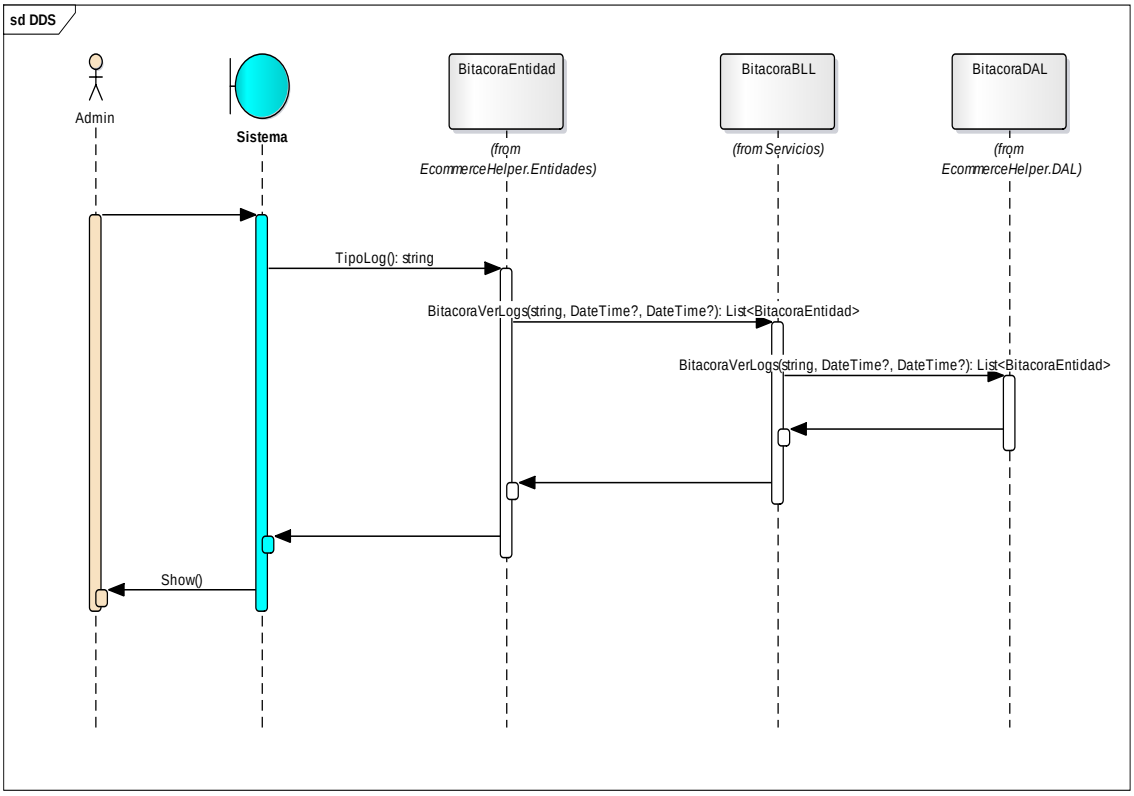

class EcommerceHelper
EcommerceHelper.Entidades::BitacoraEntidad
«property»
+ Accion(): string
+ Fecha(): DateTime
+ IdBitacoraLog(): int
+ IdUsuario(): int
+ Mensaje(): string
+ NombreUsuario(): string
+ TipoLog(): string
«use»
Servicios::BitacoraBLL
- GestorBitacora: BitacoraDAL = new BitacoraDAL()
+ BitacoraVerLogs(string, DateTime?, DateTime?): List<BitacoraEntidad>

Diagrama de Secuencia CU - 026 – Ver Logs



Pantalla

Logs del sistema
Evento
Desde
27/11/2020
Hasta
27/11/2020
Buscar
NombreUsuario
Fecha
Tipo
Accion
Mensaje
Gonzalez
27/11/2020
Evento
Logueo
Logueo Correcto
Gonzalez
27/11/2020
Evento
Logueo
Logueo Correcto

CU – 038 – Genera Digito Verificador Vertical

Precondición:
Ingresar al sistema.

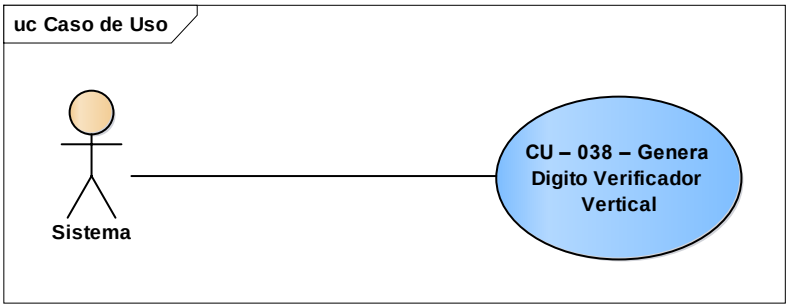
Actores primarios:
sistema.

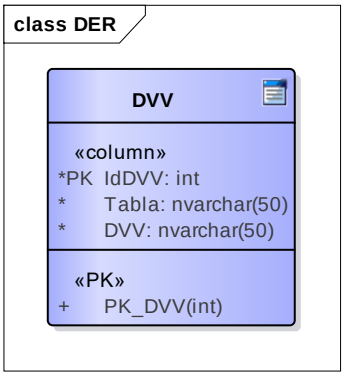
Camino principal del caso de uso – Genera DVV

1. En el caso que el sistema quiera calcular un digito verificador vertical.
2. El sistema obtiene el digito verificador de cada tabla.
3. El sistema escribe en la tabla DVV de la base de datos el valor del digito vertical calculado para la tabla afectada.

Caminos Alternativos
No existen caminos alternativos

Postcondición:
El sistema calcula con éxito el digito verificador


uc Caso de Uso
Sistema
CU – 038 – Genera Digito Verificador Vertical


class DER
DVV
«column»
*PK IdDVV: int
* Tabla: nvarchar(50)
* DVV: nvarchar(50)
«PK»
+ PK_DVV(int)

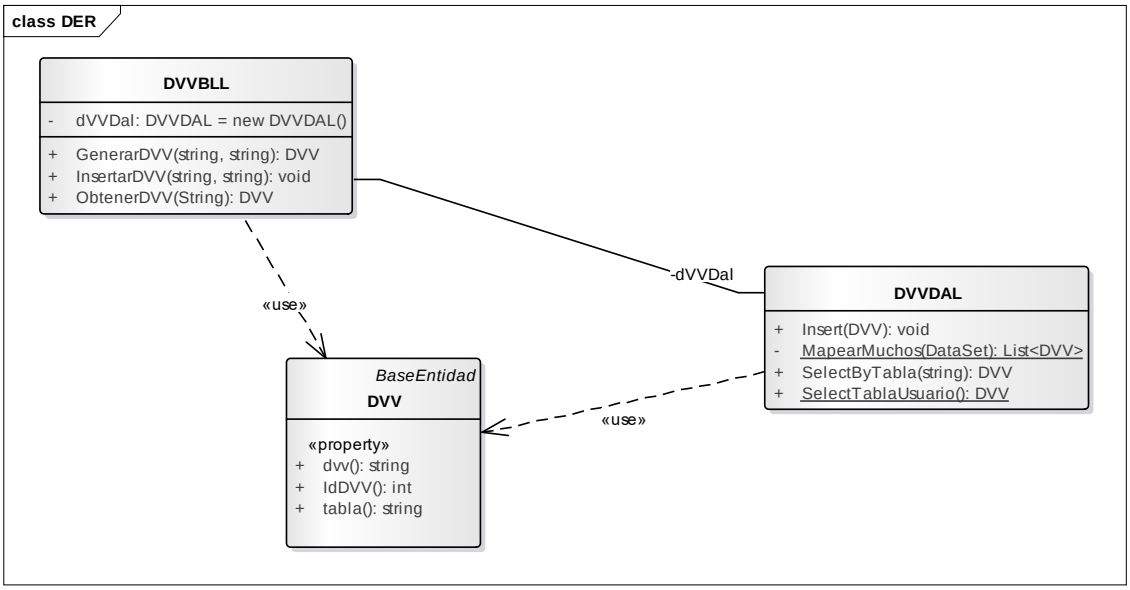
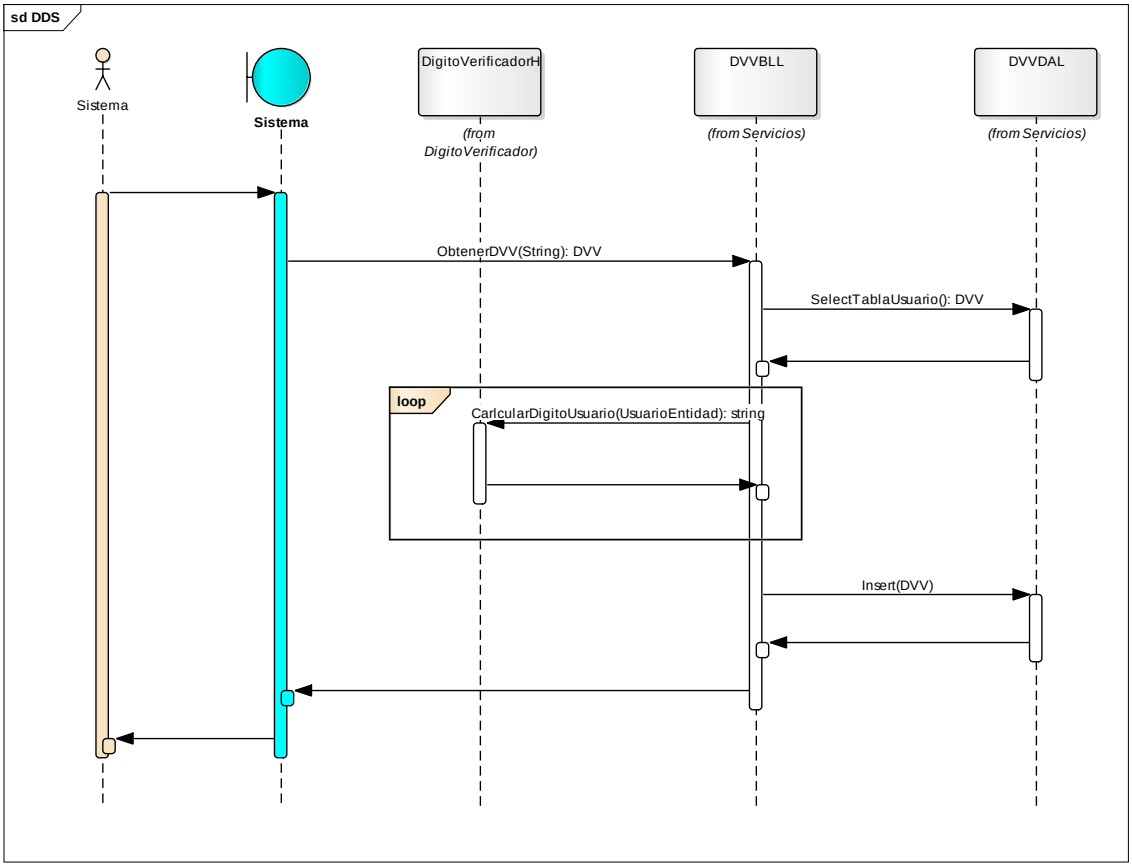

class DER
DVVBLL
- dVVDal: DVVDAL = new DVVDAL()
+ GenerarDVV(string, string): DVV
+ InsertarDVV(string, string): void
+ ObtenerDVV(String): DVV
-dVVDal
«use»
DVVDAL
+ Insert(DVV): void
- MapearMuchos(DataSet): List<DVV>
+ SelectByTabla(string): DVV
+ SelectTablaUsuario(): DVV
«use»
BaseEntidad
DVV
«property»
+ dvv(): string
+ IdDVV(): int
+ tabla(): string

Diagrama de Secuencia 038 – Genera Digito Verificador Vertical



CU – 39 – Generar Digito Verificador Horizontal

Precondición:
Ingresar al sistema.

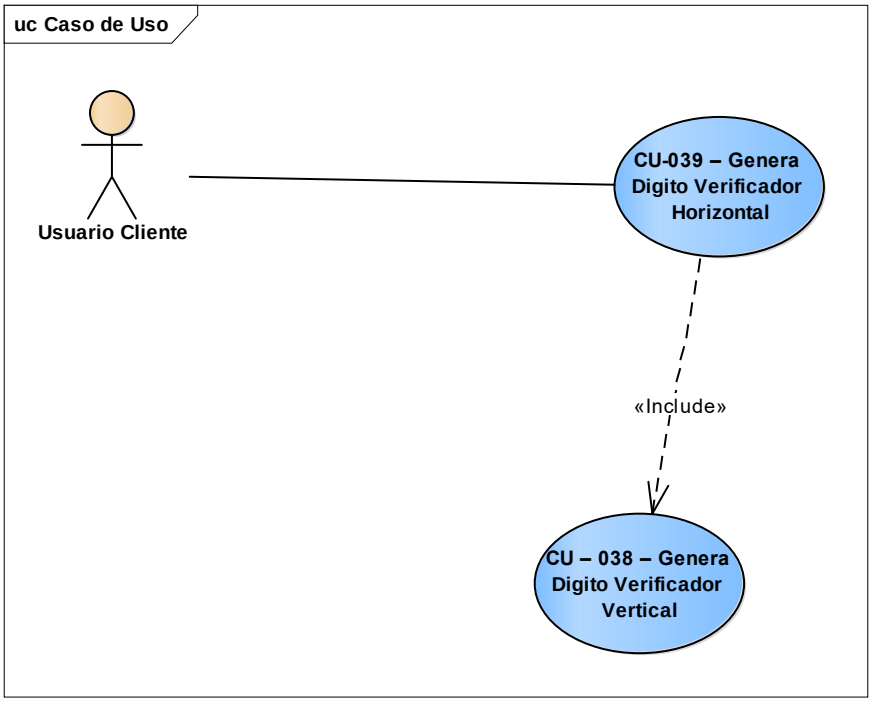
Actores primarios:
sistema.

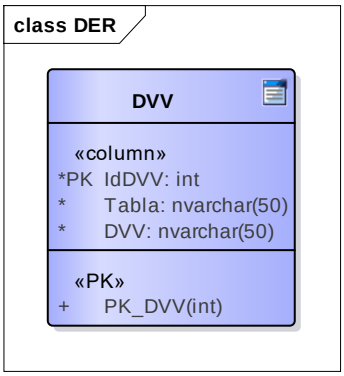
Camino principal del caso de uso – Genera DVH

1. En el caso que el sistema quiera calcular un digito verificador Horizontal.
2. El sistema calcula el valor del digito verificador horizontal para cada registro.
3. El sistema registra el valor calculado para cada registro en la columna DVH.
4. El sistema calcula el valor del digito verificador vertical para cada dicha tabla.
5. El sistema escribe en la tabla DVV de la base de datos el valor del digito vertical calculado para la tabla afectada.

Caminos Alternativos
No existen caminos alternativos

Postcondición:
El sistema calcula con éxito el digito verificador


uc Caso de Uso
Usuario Cliente
CU-039 – Genera Digito Verificador Horizontal
«Include»
CU – 038 – Genera Digito Verificador Vertical


class DER
DVV
«column»
*PK IdDVV: int
* Tabla: nvarchar(50)
* DVV: nvarchar(50)
«PK»
+ PK_DVV(int)

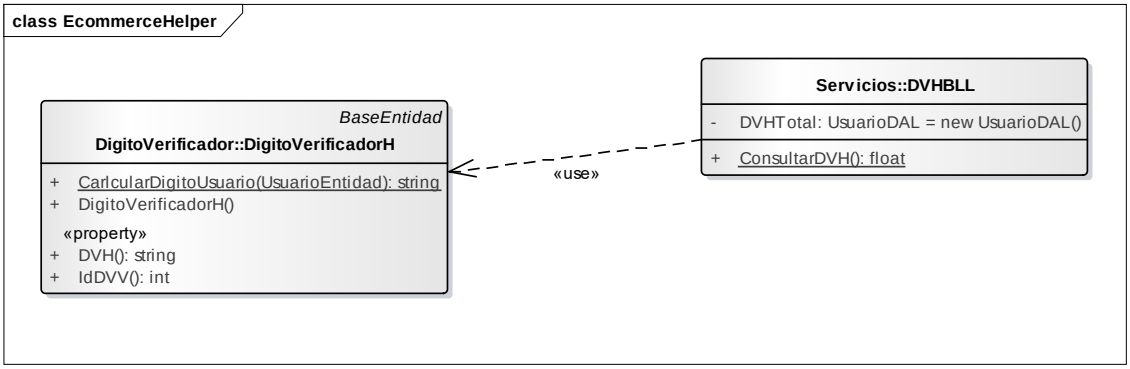
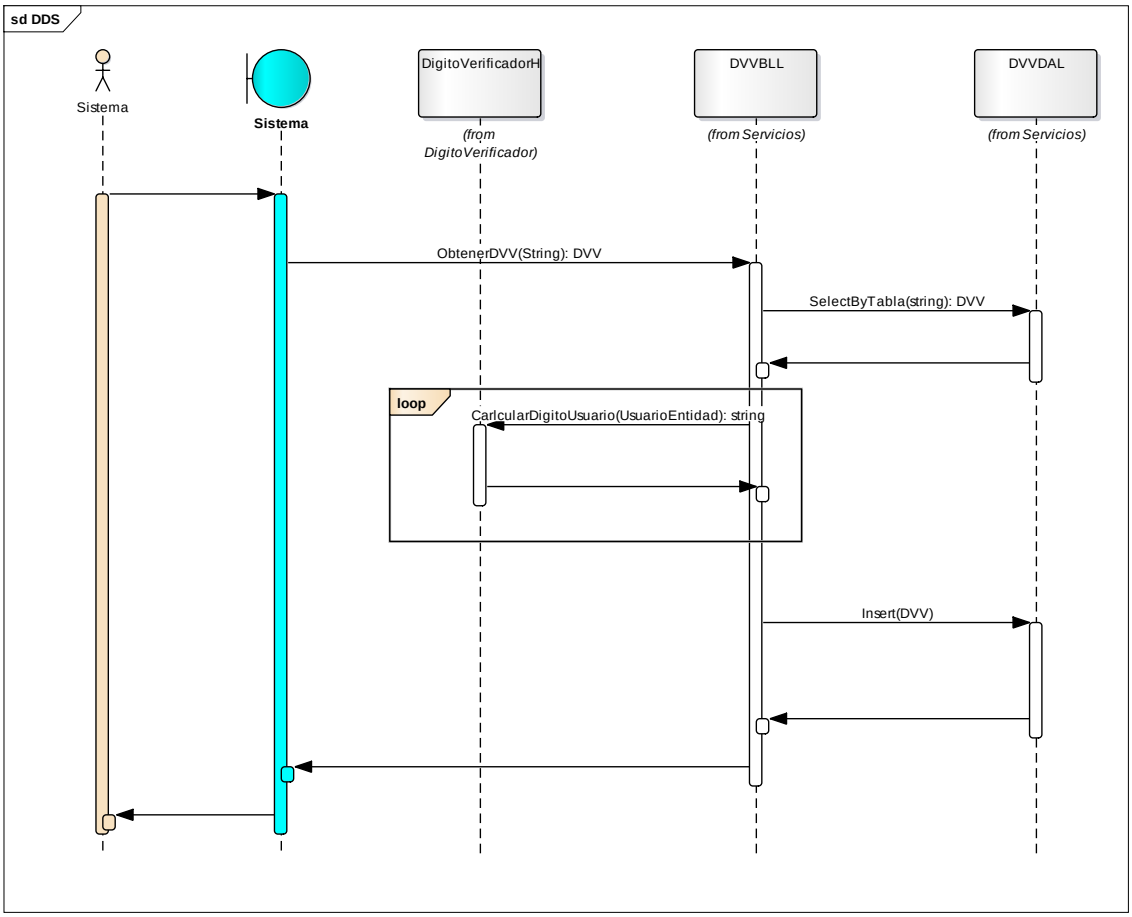

class EcommerceHelper
BaseEntidad
DigitoVerificador::DigitoVerificadorH
+ CarlcularDigitoUsuario(UsuarioEntidad): string
+ DigitoVerificadorH()
«property»
+ DVH(): string
+ IdDVV(): int
«use»
Servicios::DVHBLL
- DVHTotal: UsuarioDAL = new UsuarioDAL()
+ ConsultarDVH(): float

Diagrama de Secuencia CU – 39 – Generar Digito Verificador Horizontal



CU – 43 – Modificar permisos a los Roles

Precondición:
El administrador debe estar logueado.

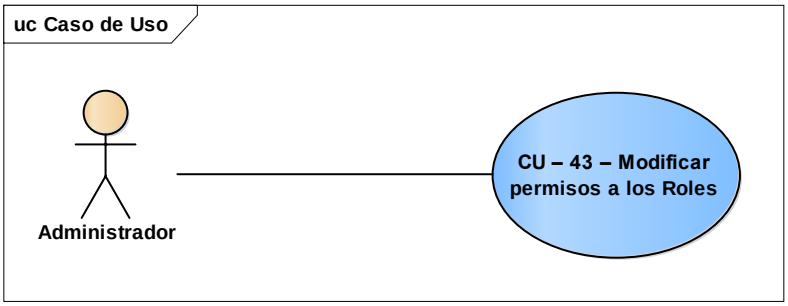
Actores primarios:
Administrador.

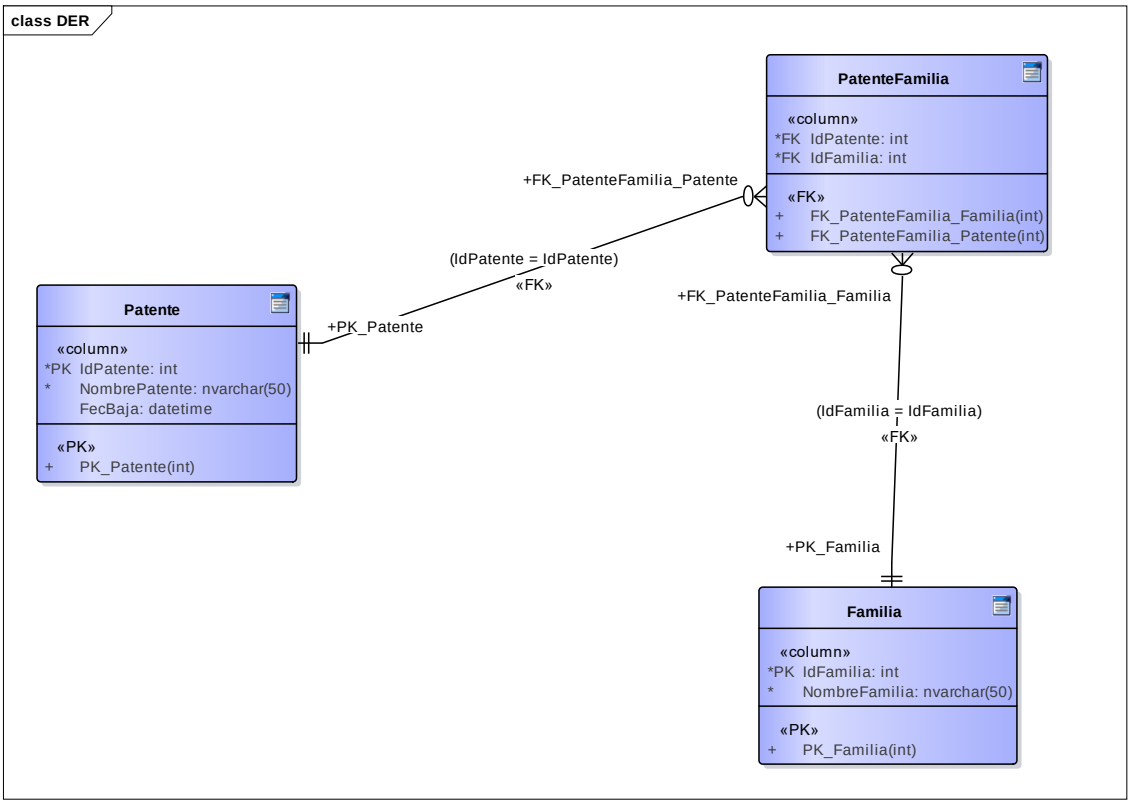
Camino principal del caso de uso – Modificar permisos a los Roles

1. El Administrador se dirige a la sección de la Intranet “Menú Administración – ABM Permisos”
2. El Sistema redirige a la pantalla “ABM Permisos” y selecciona un rol / familia a la cual le desea modificar permisos / patentes.
3. El sistema despliega los permisos / patentes disponibles de agregar.
4. El Administrador selecciona los permisos / patentes que desea modificar al rol / familia.
5. El administrador selecciona guardar modificaciones.
6. El sistema guarda los cambios en la base de datos.

Caminos Alternativos
No existen caminos alternativos

Postcondición:
Quedan asignados los nuevos permisos al rol.


uc Caso de Uso
Administrador
CU – 43 – Modificar permisos a los Roles


class DER
PatenteFamilia
«column»
*FK IdPatente: int
*FK IdFamilia: int
«FK»
+ FK_PatenteFamilia_Familia(int)
+ FK_PatenteFamilia_Patente(int)
+FK_PatenteFamilia_Patente
(IdPatente = IdPatente)
«FK»
+FK_PatenteFamilia_Familia
Patente
«column»
*PK IdPatente: int
* NombrePatente: nvarchar(50)
FecBaja: datetime
«PK»
+ PK_Patente(int)
+PK_Patente
(IdFamilia = IdFamilia)
«FK»
+PK_Familia
Familia
«column»
*PK IdFamilia: int
* NombreFamilia: nvarchar(50)
«PK»
+ PK_Familia(int)

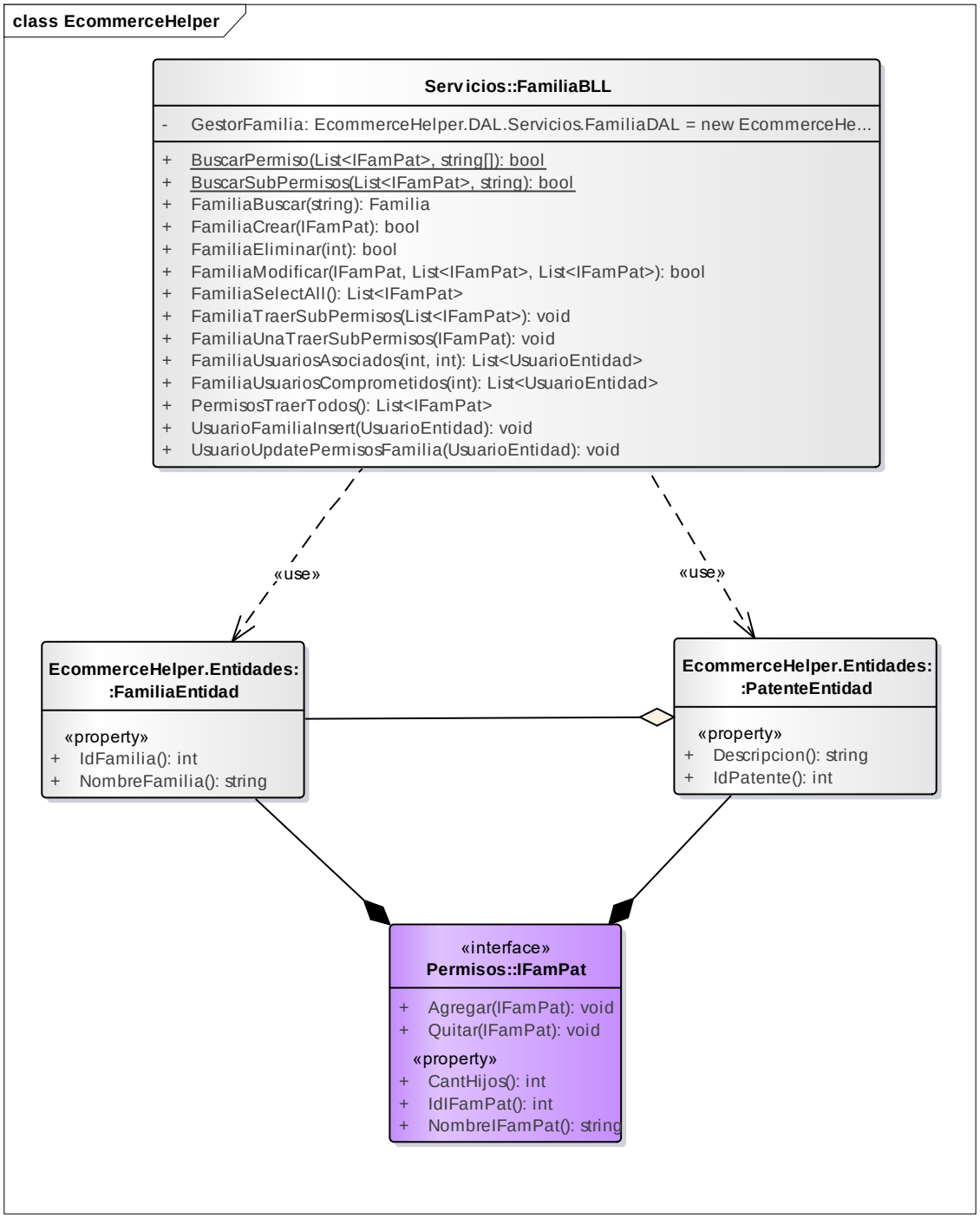
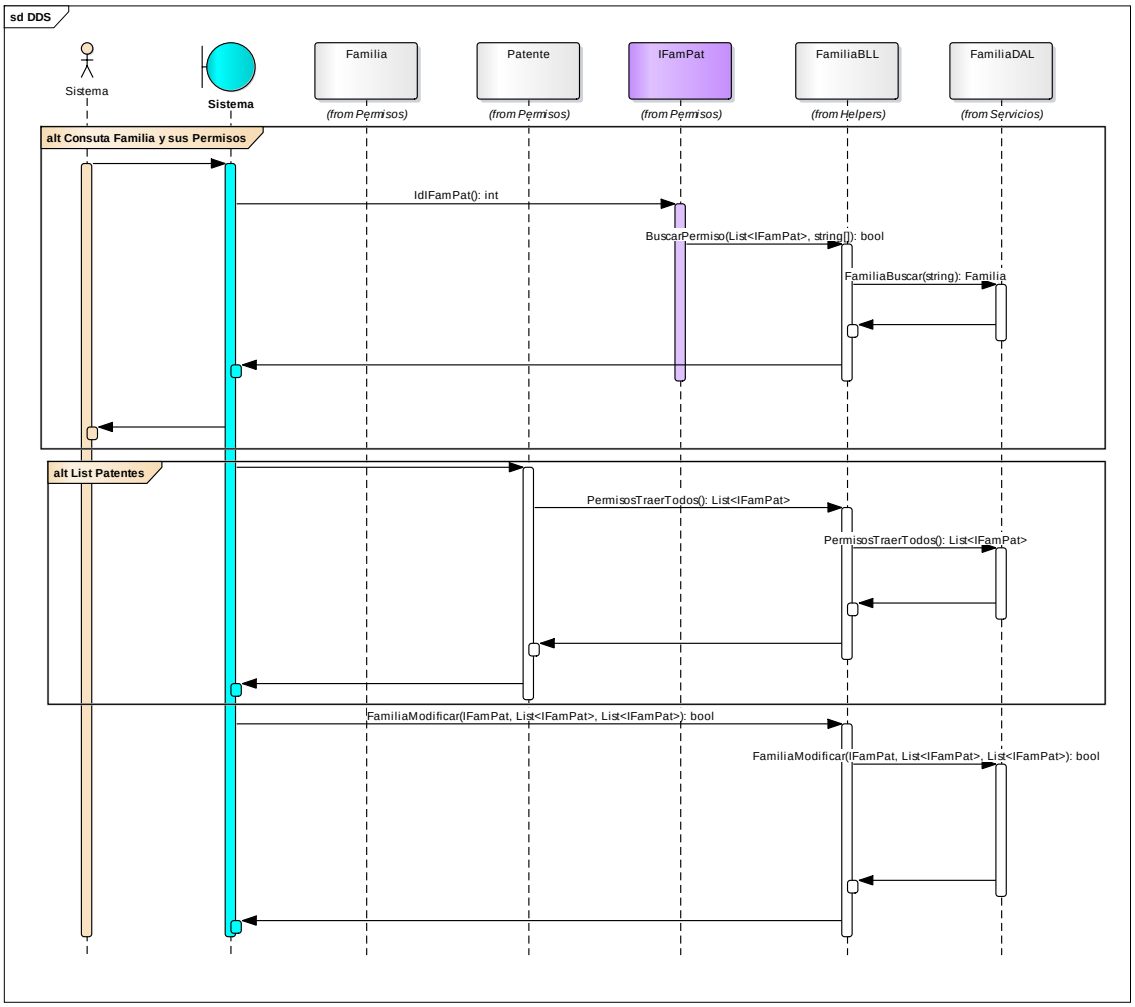
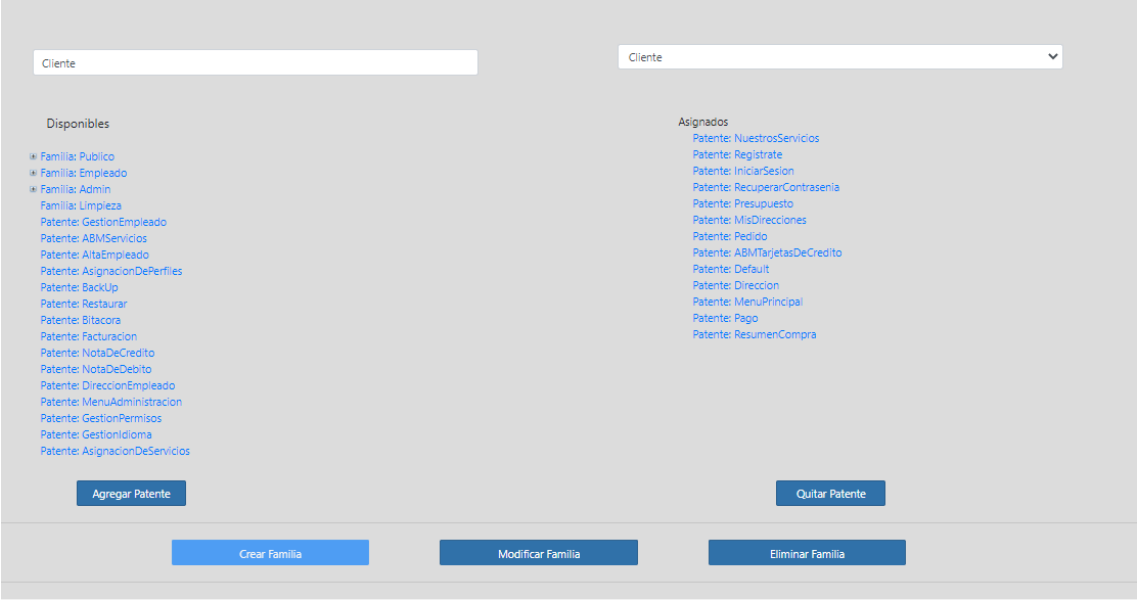

class EcommerceHelper
Servicios::FamiliaBLL
- GestorFamilia: EcommerceHelper.DAL.Servicios.FamiliaDAL = new EcommerceHe...
+ BuscarPermiso(List<IFamPat>, string[]): bool
+ BuscarSubPermisos(List<IFamPat>, string): bool
+ FamiliaBuscar(string): Familia
+ FamiliaCrear(IFamPat): bool
+ FamiliaEliminar(int): bool
+ FamiliaModificar(IFamPat, List<IFamPat>, List<IFamPat>): bool
+ FamiliaSelectAll(): List<IFamPat>
+ FamiliaTraerSubPermisos(List<IFamPat>): void
+ FamiliaUnaTraerSubPermisos(IFamPat): void
+ FamiliaUsuariosAsociados(int, int): List<UsuarioEntidad>
+ FamiliaUsuariosComprometidos(int): List<UsuarioEntidad>
+ PermisosTraerTodos(): List<IFamPat>
+ UsuarioFamiliaInsert(UsuarioEntidad): void
+ UsuarioUpdatePermisosFamilia(UsuarioEntidad): void
«use»
«use»
EcommerceHelper.Entidades::FamiliaEntidad
«property»
+ IdFamilia(): int
+ NombreFamilia(): string
EcommerceHelper.Entidades::PatenteEntidad
«property»
+ Descripcion(): string
+ IdPatente(): int
«interface»
Permisos::IFamPat
+ Agregar(IFamPat): void
+ Quitar(IFamPat): void
«property»
+ CantHijos(): int
+ IdIFamPat(): int
+ NombreIFamPat(): string

Diagrama de Secuencia CU – 43 – Asignar permisos a los Roles



Pantalla


Cliente
Cliente
Disponibles
Familia: Publico
Familia: Empleado
Familia: Admin
Familia: Limpieza
Patente: GestionEmpleado
Patente: ABMServicios
Patente: AltaEmpleado
Patente: AsignacionDePerfiles
Patente: BackUp
Patente: Restaurar
Patente: Bitacora
Patente: Facturacion
Patente: NotaDeCredito
Patente: NotaDeDebito
Patente: DireccionEmpleado
Patente: MenuAdministracion
Patente: GestionPermisos
Patente: GestionIdioma
Patente: AsignacionDeServicios
Asignados
Patente: NuestrosServicios
Patente: Registrate
Patente: IniciarSesion
Patente: RecuperarContrasenia
Patente: Presupuesto
Patente: MisDirecciones
Patente: Pedido
Patente: ABMTarjetasDeCredito
Patente: Default
Patente: Direccion
Patente: MenuPrincipal
Patente: Pago
Patente: ResumenCompra
Agregar Patente
Quitar Patente
Crear Familia
Modificar Familia
Eliminar Familia

CU – O45 – Consultar Roles

Precondición:
El administrador debe estar logueado.

Actores primarios:
Administrador.

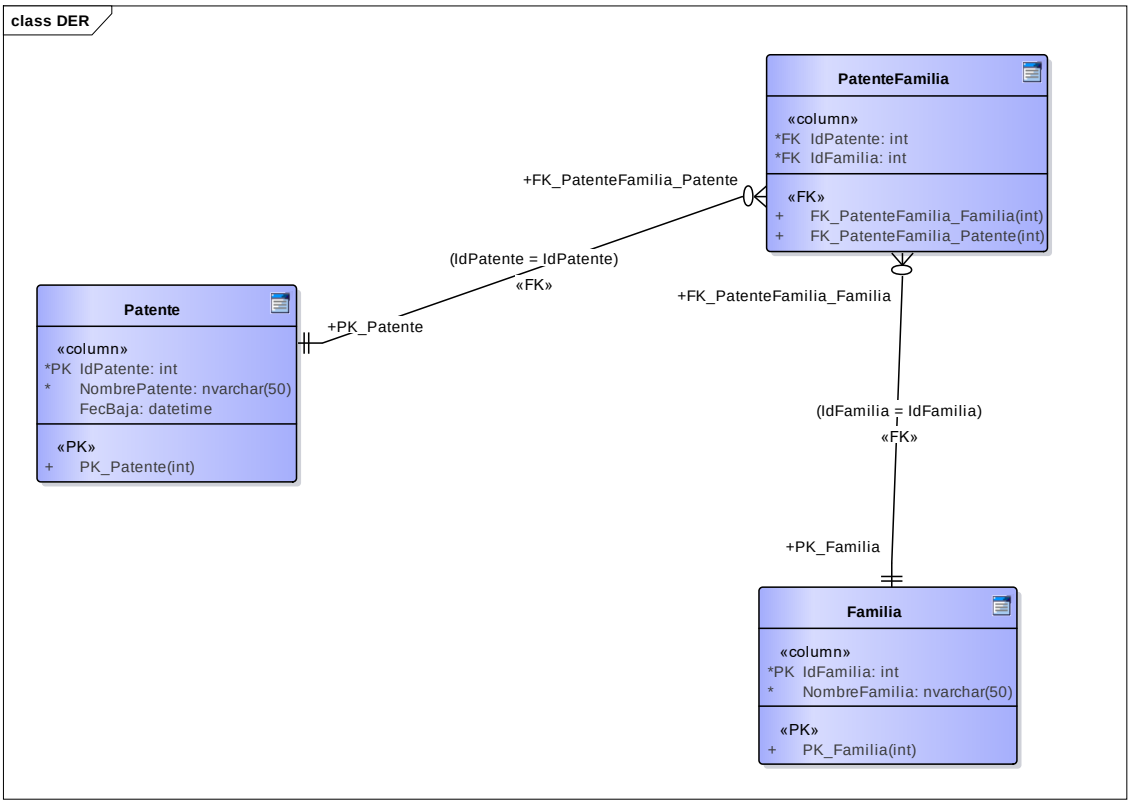
Camino principal del caso de uso – Consultar permisos según Roles

1. El Administrador se dirige a la sección de la Intranet “Menú Administración – ABM Permisos”
2. El Sistema carga en una lista desplegable los roles/ familias disponibles.
3. El Administrador selecciona el rol / familia.
4. El sistema despliega los permisos asociados a ese rol seleccionado.

Caminos Alternativos
No existen caminos alternativos

Postcondición:
Quedan asignados los nuevos permisos al rol.

uc Caso de Uso
Administrador
CU – 44 – Consultar Roles


class DER
PatenteFamilia
«column»
*FK IdPatente: int
*FK IdFamilia: int
«FK»
+ FK_PatenteFamilia_Familia(int)
+ FK_PatenteFamilia_Patente(int)
+FK_PatenteFamilia_Patente
(IdPatente = IdPatente)
«FK»
+FK_PatenteFamilia_Familia
Patente
«column»
*PK IdPatente: int
* NombrePatente: nvarchar(50)
FecBaja: datetime
«PK»
+ PK_Patente(int)
+PK_Patente
(IdFamilia = IdFamilia)
«FK»
+PK_Familia
Familia
«column»
*PK IdFamilia: int
* NombreFamilia: nvarchar(50)
«PK»
+ PK_Familia(int)

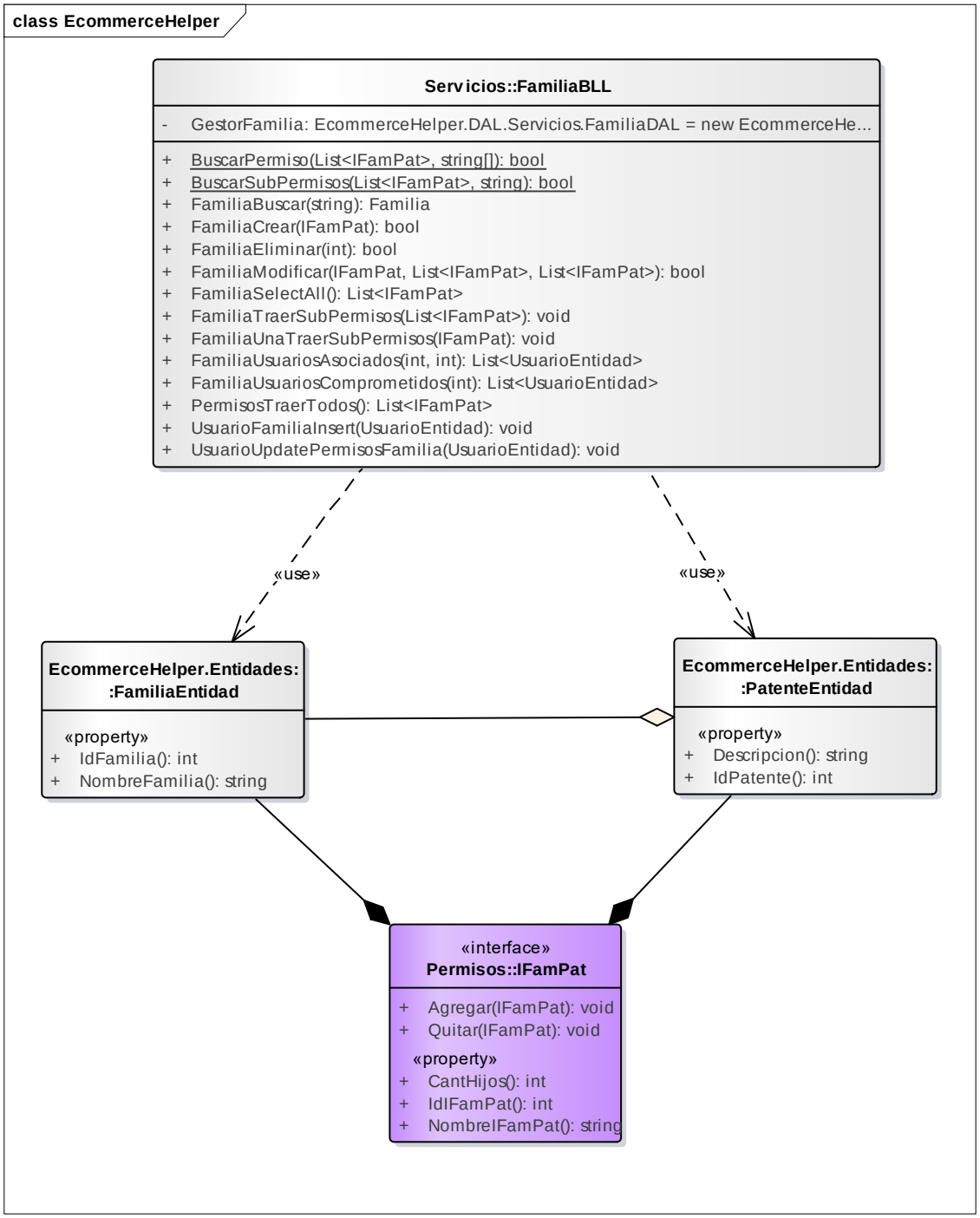
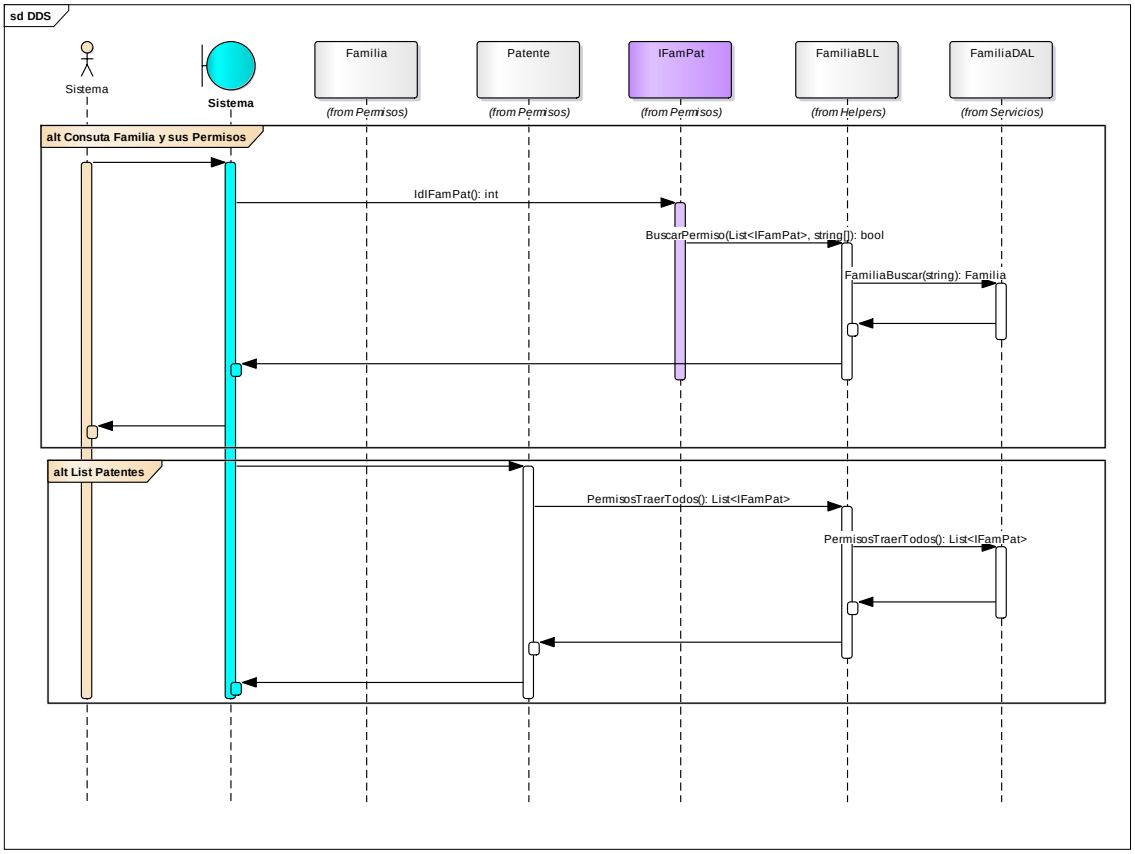
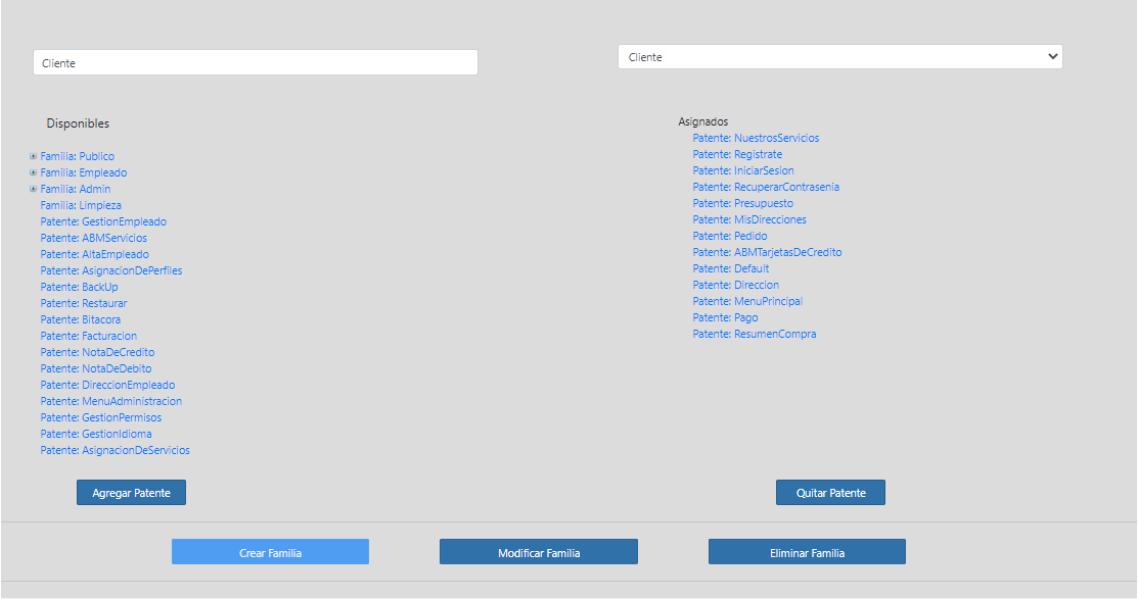

class EcommerceHelper
Servicios::FamiliaBLL
- GestorFamilia: EcommerceHelper.DAL.Servicios.FamiliaDAL = new EcommerceHe...
+ BuscarPermiso(List<IFamPat>, string[]): bool
+ BuscarSubPermisos(List<IFamPat>, string): bool
+ FamiliaBuscar(string): Familia
+ FamiliaCrear(IFamPat): bool
+ FamiliaEliminar(int): bool
+ FamiliaModificar(IFamPat, List<IFamPat>, List<IFamPat>): bool
+ FamiliaSelectAll(): List<IFamPat>
+ FamiliaTraerSubPermisos(List<IFamPat>): void
+ FamiliaUnaTraerSubPermisos(IFamPat): void
+ FamiliaUsuariosAsociados(int, int): List<UsuarioEntidad>
+ FamiliaUsuariosComprometidos(int): List<UsuarioEntidad>
+ PermisosTraerTodos(): List<IFamPat>
+ UsuarioFamiliaInsert(UsuarioEntidad): void
+ UsuarioUpdatePermisosFamilia(UsuarioEntidad): void
«use»
«use»
EcommerceHelper.Entidades::FamiliaEntidad
«property»
+ IdFamilia(): int
+ NombreFamilia(): string
EcommerceHelper.Entidades::PatenteEntidad
«property»
+ Descripcion(): string
+ IdPatente(): int
«interface»
Permisos::IFamPat
+ Agregar(IFamPat): void
+ Quitar(IFamPat): void
«property»
+ CantHijos(): int
+ IdIFamPat(): int
+ NombreIFamPat(): string

Diagrama de Secuencia CU – 44 – Consultar Roles



Pantalla


Cliente
Cliente
Disponibles
Familia: Publico
Familia: Empleado
Familia: Admin
Familia: Limpieza
Patente: GestionEmpleado
Patente: ABMServicios
Patente: AltaEmpleado
Patente: AsignacionDePerfiles
Patente: BackUp
Patente: Restaurar
Patente: Bitacora
Patente: Facturacion
Patente: NotaDeCredito
Patente: NotaDeDebito
Patente: DireccionEmpleado
Patente: MenuAdministracion
Patente: GestionPermisos
Patente: GestionIdioma
Patente: AsignacionDeServicios
Asignados
Patente: NuestrosServicios
Patente: Registrate
Patente: IniciarSesion
Patente: RecuperarContrasenia
Patente: Presupuesto
Patente: MisDirecciones
Patente: Pedido
Patente: ABMTarjetasDeCredito
Patente: Default
Patente: Direccion
Patente: MenuPrincipal
Patente: Pago
Patente: ResumenCompra
Agregar Patente
Quitar Patente
Crear Familia
Modificar Familia
Eliminar Familia

10.11 DER (Diagrama Entidad Relación)

DER Negocio en página 409

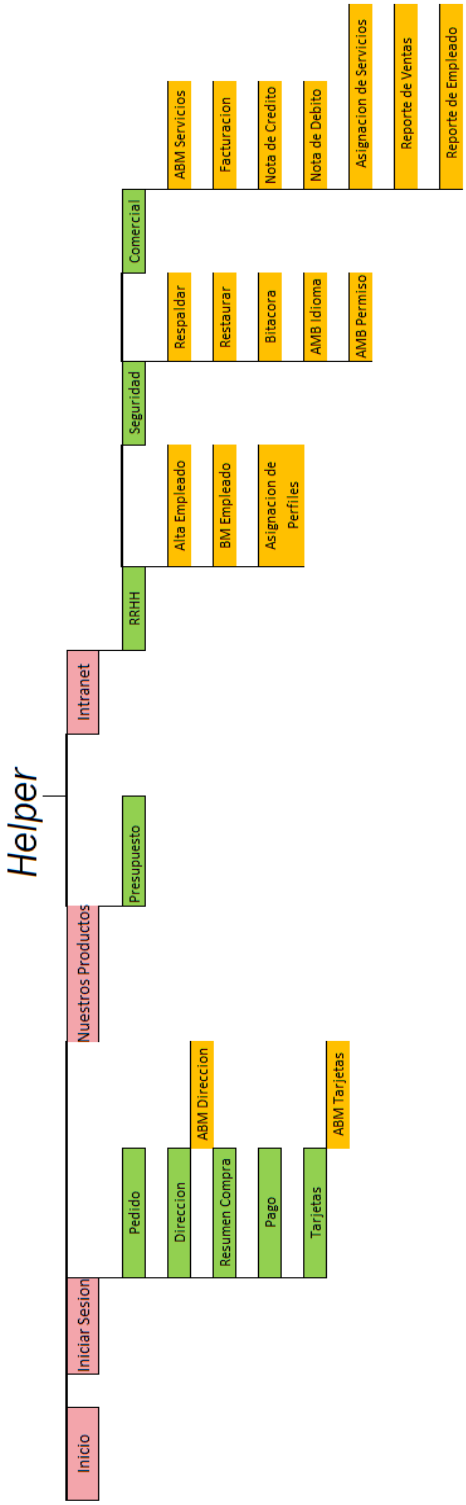
10.13 Diagrama de Clases

Diagrama de Clases en página 410

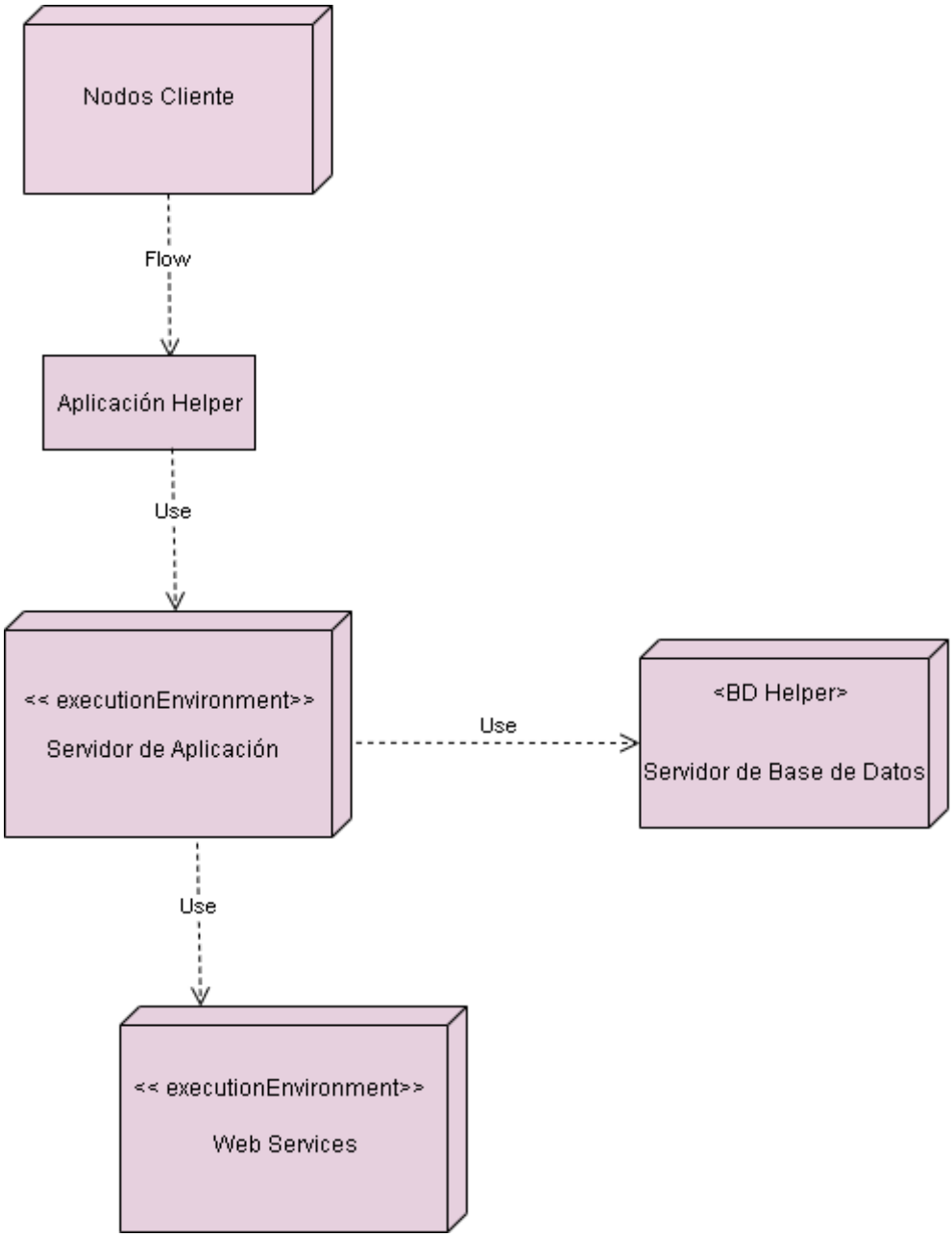
10.14 Diagrama de Contexto

Diagrama de Clases en página 411

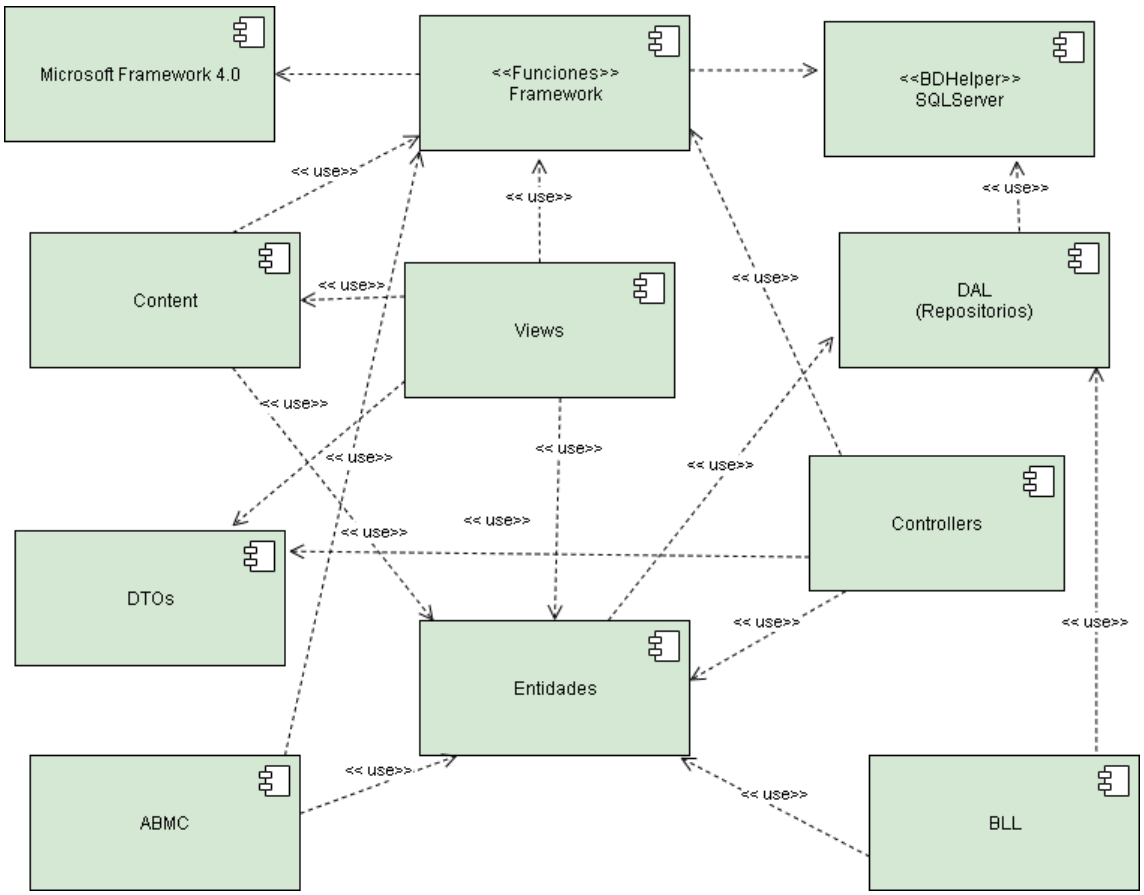
10.16 Mapa de Navegación



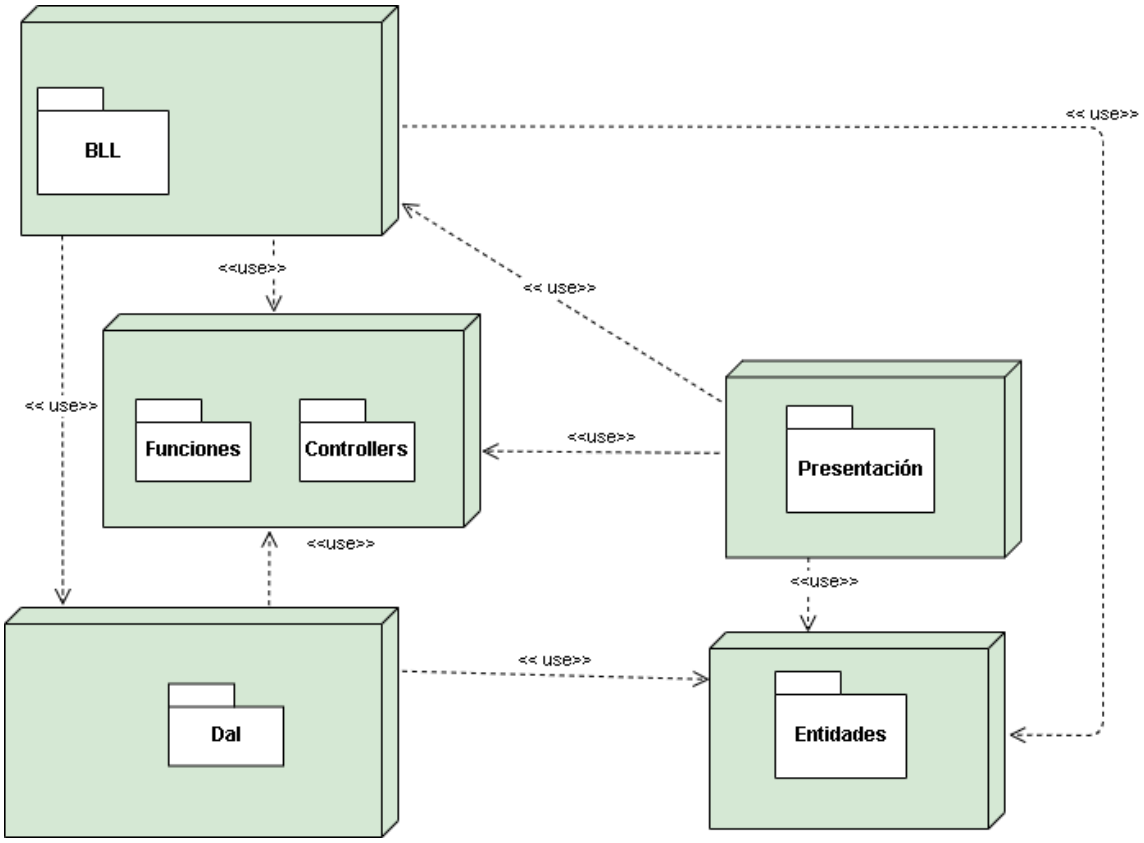
10.16 Diagrama de Despliegue

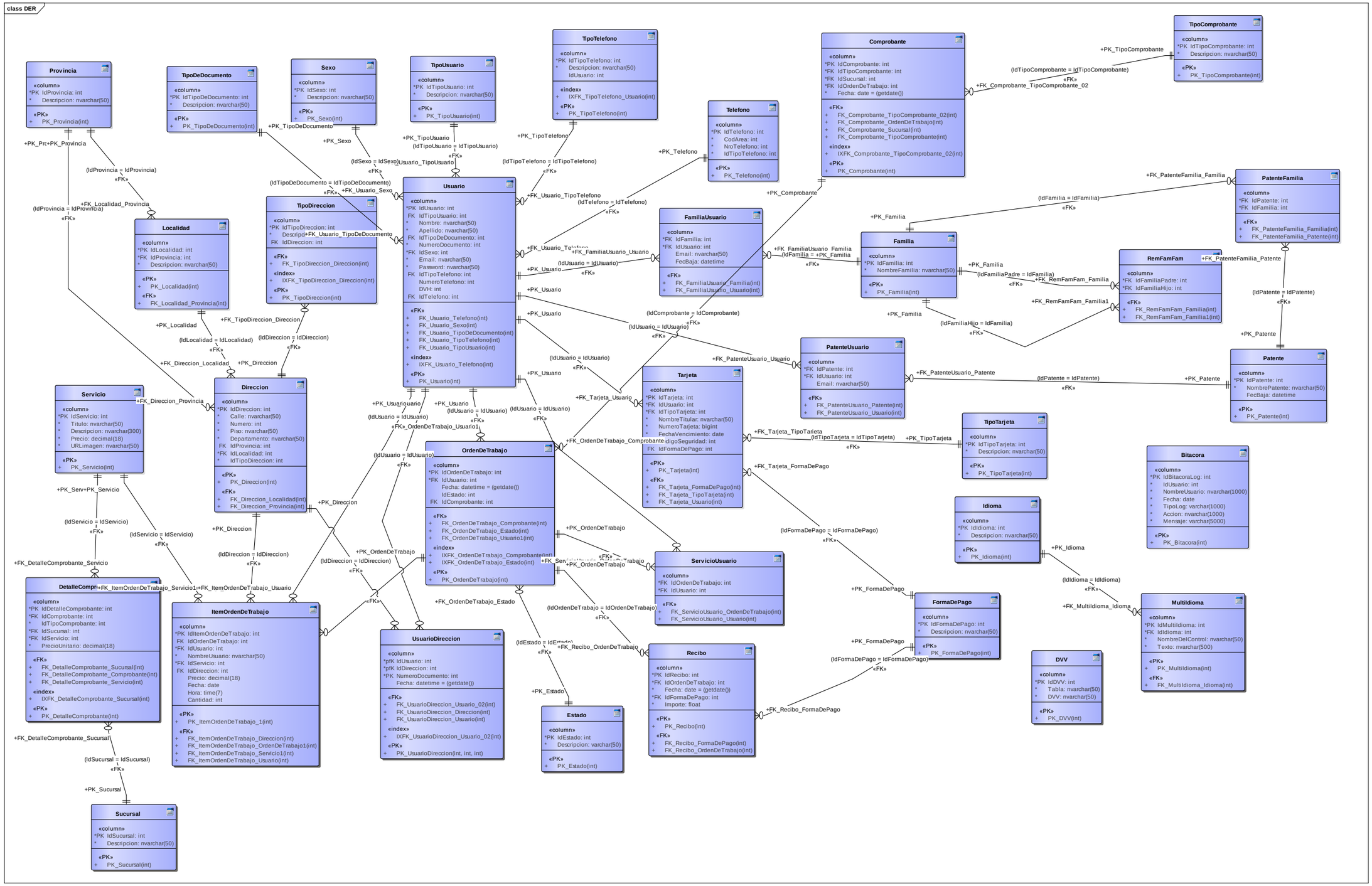


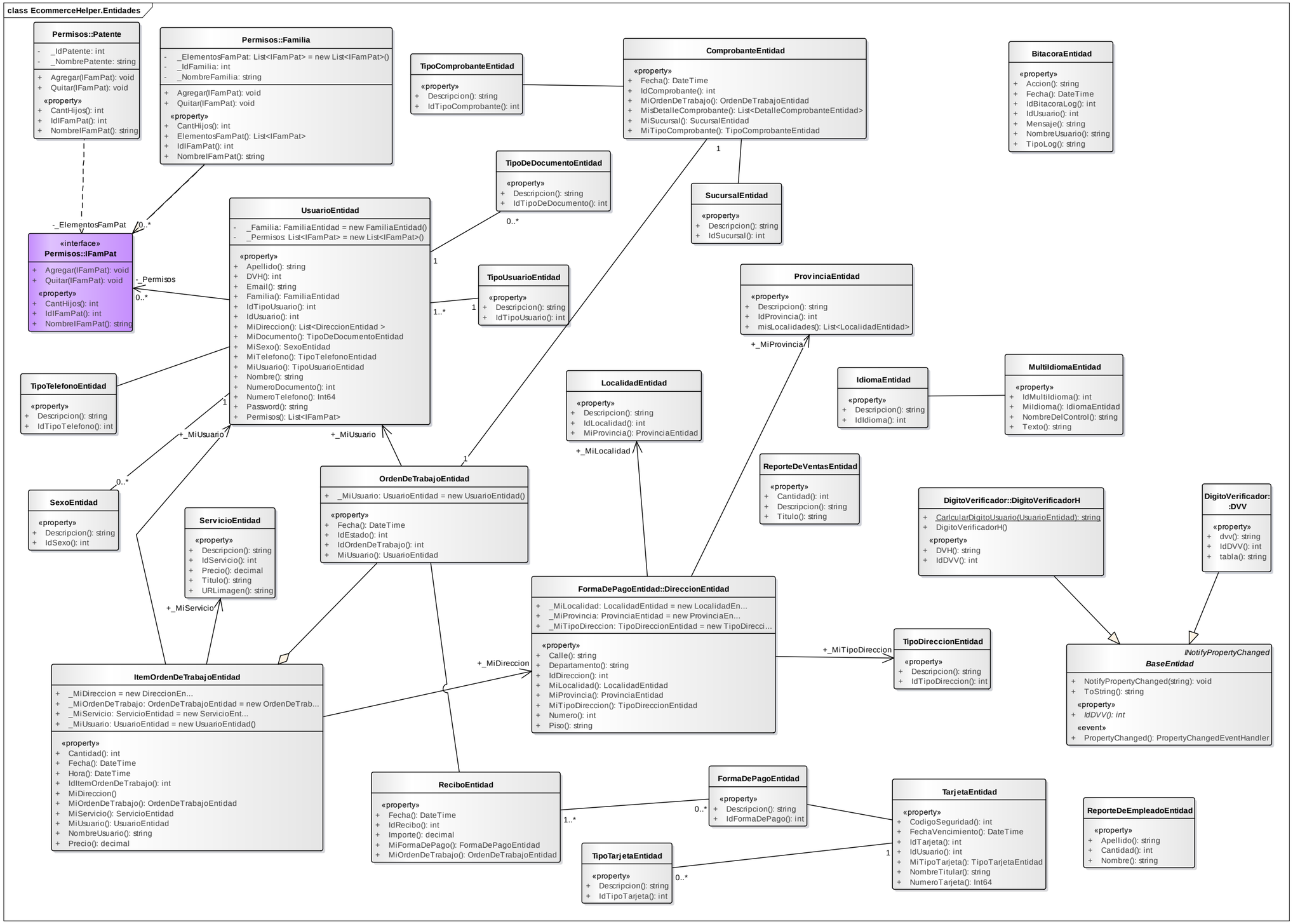
10.17 Diagrama de Componentes

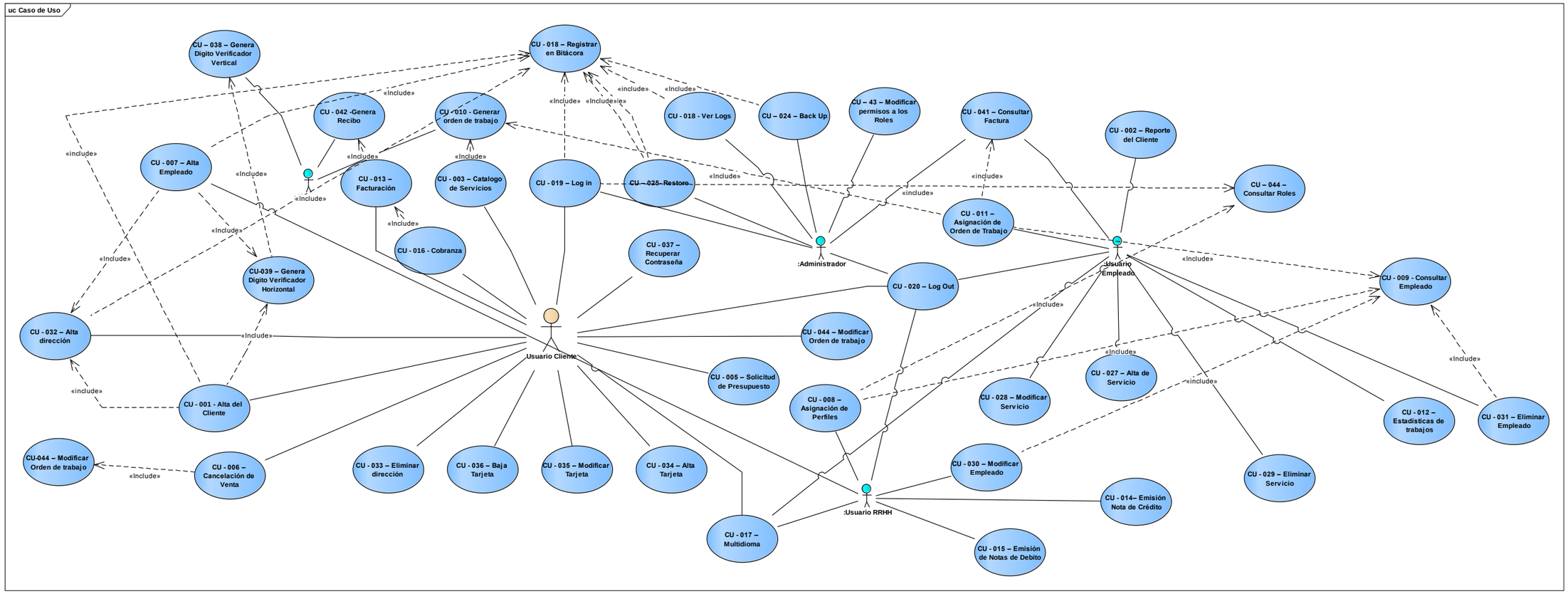


10.18 Diagrama de Paquetes




class DER
Provincia
«column»
*PK IdProvincia: int
* Descripcion: nvarchar(50)
«PK»
+ PK_Provincia(int)
TipoDeDocumento
«column»
*PK IdTipoDeDocumento: int
* Descripcion: nvarchar(50)
«PK»
+ PK_TipoDeDocumento(int)
Sexo
«column»
*PK IdSexo: int
* Descripcion: nvarchar(50)
«PK»
+ PK_Sexo(int)
TipoUsuario
«column»
*PK IdTipoUsuario: int
* Descripcion: nvarchar(50)
«PK»
+ PK_TipoUsuario(int)
TipoTelefono
«column»
*PK IdTipoTelefono: int
* Descripcion: nvarchar(50)
IdUsuario: int
«index»
+ IXFK_TipoTelefono_Usuario(int)
«PK»
+ PK_TipoTelefono(int)
Comprobante
«column»
*PK IdComprobante: int
*FK IdTipoComprobante: int
*FK IdSucursal: int
*FK IdOrdenDeTrabajo: int
* Fecha: date = (getdate())
«FK»
+ FK_Comprobante_TipoComprobante_02(int)
+ FK_Comprobante_OrdenDeTrabajo(int)
+ FK_Comprobante_Sucursal(int)
+ FK_Comprobante_TipoComprobante(int)
«index»
+ IXFK_Comprobante_TipoComprobante_02(int)
«PK»
+ PK_Comprobante(int)
TipoComprobante
«column»
*PK IdTipoComprobante: int
* Descripcion: nvarchar(50)
«PK»
+ PK_TipoComprobante(int)
Telefono
«column»
*PK IdTelefono: int
* CodArea: int
* NroTelefono: int
IdTipoTelefono: int
«PK»
+ PK_Telefono(int)
Localidad
«column»
*PK IdLocalidad: int
*FK IdProvincia: int
* Descripcion: nvarchar(50)
«PK»
+ PK_Localidad(int)
«FK»
+ FK_Localidad_Provincia(int)
TipoDireccion
«column»
*PK IdTipoDireccion: int
* Descripcion: nvarchar(50)
FK IdDireccion: int
«FK»
+ FK_TipoDireccion_Direccion(int)
«index»
+ IXFK_TipoDireccion_Direccion(int)
«PK»
+ PK_TipoDireccion(int)
Usuario
«column»
*PK IdUsuario: int
FK IdTipoUsuario: int
* Nombre: nvarchar(50)
* Apellido: nvarchar(50)
FK IdTipoDeDocumento: int
* NumeroDocumento: int
*FK IdSexo: int
* Email: nvarchar(50)
* Password: nvarchar(50)
FK IdTipoTelefono: int
NumeroTelefono: int
DVH: int
FK IdTelefono: int
«FK»
+ FK_Usuario_Telefono(int)
+ FK_Usuario_Sexo(int)
+ FK_Usuario_TipoDeDocumento(int)
+ FK_Usuario_TipoTelefono(int)
+ FK_Usuario_TipoUsuario(int)
«index»
+ IXFK_Usuario_Telefono(int)
«PK»
+ PK_Usuario(int)
FamiliaUsuario
«column»
*FK IdFamilia: int
*FK IdUsuario: int
Email: nvarchar(50)
FecBaja: datetime
«FK»
+ FK_FamiliaUsuario_Familia(int)
+ FK_FamiliaUsuario_Usuario(int)
Familia
«column»
*PK IdFamilia: int
* NombreFamilia: nvarchar(50)
«PK»
+ PK_Familia(int)
PatenteFamilia
«column»
*FK IdPatente: int
*FK IdFamilia: int
«FK»
+ FK_PatenteFamilia_Familia(int)
+ FK_PatenteFamilia_Patente(int)
RemFamFam
«column»
*FK IdFamiliaPadre: int
*FK IdFamiliaHijo: int
«FK»
+ FK_RemFamFam_Familia(int)
+ FK_RemFamFam_Familia1(int)
PatenteUsuario
«column»
*FK IdPatente: int
*FK IdUsuario: int
Email: nvarchar(50)
«FK»
+ FK_PatenteUsuario_Patente(int)
+ FK_PatenteUsuario_Usuario(int)
Patente
«column»
*PK IdPatente: int
* NombrePatente: nvarchar(50)
FecBaja: datetime
«PK»
+ PK_Patente(int)
Servicio
«column»
*PK IdServicio: int
* Titulo: nvarchar(50)
* Descripcion: nvarchar(300)
* Precio: decimal(18)
* URLimagen: nvarchar(50)
«PK»
+ PK_Servicio(int)
Direccion
«column»
*PK IdDireccion: int
* Calle: nvarchar(50)
* Numero: int
* Piso: nvarchar(50)
* Departamento: nvarchar(50)
FK IdProvincia: int
*FK IdLocalidad: int
* IdTipoDireccion: int
«PK»
+ PK_Direccion(int)
«FK»
+ FK_Direccion_Localidad(int)
+ FK_Direccion_Provincia(int)
Tarjeta
«column»
*PK IdTarjeta: int
*FK IdUsuario: int
*FK IdTipoTarjeta: int
* NombreTitular: nvarchar(50)
* NumeroTarjeta: bigint
* FechaVencimiento: date
CodigoSeguridad: int
FK IdFormaDePago: int
«PK»
+ PK_Tarjeta(int)
«FK»
+ FK_Tarjeta_FormaDePago(int)
+ FK_Tarjeta_TipoTarjeta(int)
+ FK_Tarjeta_Usuario(int)
TipoTarjeta
«column»
*PK IdTipoTarjeta: int
* Descripcion: nvarchar(50)
«PK»
+ PK_TipoTarjeta(int)
Bitacora
«column»
*PK IdBitacoraLog: int
* IdUsuario: int
* NombreUsuario: nvarchar(1000)
* Fecha: date
* TipoLog: varchar(1000)
* Accion: nvarchar(1000)
* Mensaje: varchar(5000)
«PK»
+ PK_Bitacora(int)
Idioma
«column»
*PK IdIdioma: int
* Descripcion: nvarchar(50)
«PK»
+ PK_Idioma(int)
DetalleComprobante
«column»
*PK IdDetalleComprobante: int
*FK IdComprobante: int
* IdTipoComprobante: int
*FK IdSucursal: int
*FK IdServicio: int
* PrecioUnitario: decimal(18)
«FK»
+ FK_DetalleComprobante_Sucursal(int)
+ FK_DetalleComprobante_Comprobante(int)
+ FK_DetalleComprobante_Servicio(int)
«index»
+ IXFK_DetalleComprobante_Sucursal(int)
«PK»
+ PK_DetalleComprobante(int)
ItemOrdenDeTrabajo
«column»
*PK IdItemOrdenDeTrabajo: int
FK IdOrdenDeTrabajo: int
*FK IdUsuario: int
* NombreUsuario: nvarchar(50)
*FK IdServicio: int
FK IdDireccion: int
Precio: decimal(18)
Fecha: date
Hora: time(7)
Cantidad: int
«PK»
+ PK_ItemOrdenDeTrabajo_1(int)
«FK»
+ FK_ItemOrdenDeTrabajo_Direccion(int)
+ FK_ItemOrdenDeTrabajo_OrdenDeTrabajo1(int)
+ FK_ItemOrdenDeTrabajo_Servicio1(int)
+ FK_ItemOrdenDeTrabajo_Usuario(int)
OrdenDeTrabajo
«column»
*PK IdOrdenDeTrabajo: int
*FK IdUsuario: int
Fecha: datetime = (getdate())
IdEstado: int
FK IdComprobante: int
«FK»
+ FK_OrdenDeTrabajo_Comprobante(int)
+ FK_OrdenDeTrabajo_Estado(int)
+ FK_OrdenDeTrabajo_Usuario1(int)
«index»
+ IXFK_OrdenDeTrabajo_Comprobante(int)
+ IXFK_OrdenDeTrabajo_Estado(int)
«PK»
+ PK_OrdenDeTrabajo(int)
ServicioUsuario
«column»
*FK IdOrdenDeTrabajo: int
*FK IdUsuario: int
«FK»
+ FK_ServicioUsuario_OrdenDeTrabajo(int)
+ FK_ServicioUsuario_Usuario(int)
UsuarioDireccion
«column»
*pfK IdUsuario: int
*pfK IdDireccion: int
*PK NumeroDocumento: int
Fecha: datetime = (getdate())
«FK»
+ FK_UsuarioDireccion_Usuario_02(int)
+ FK_UsuarioDireccion_Direccion(int)
+ FK_UsuarioDireccion_Usuario(int)
«index»
+ IXFK_UsuarioDireccion_Usuario_02(int)
«PK»
+ PK_UsuarioDireccion(int, int, int)
Estado
«column»
*PK IdEstado: int
* Descripcion: varchar(50)
«PK»
+ PK_Estado(int)
Recibo
«column»
*PK IdRecibo: int
*FK IdOrdenDeTrabajo: int
* Fecha: date = (getdate())
*FK IdFormaDePago: int
* Importe: float
«PK»
+ PK_Recibo(int)
«FK»
+ FK_Recibo_FormaDePago(int)
+ FK_Recibo_OrdenDeTrabajo(int)
FormaDePago
«column»
*PK IdFormaDePago: int
* Descripcion: nvarchar(50)
«PK»
+ PK_FormaDePago(int)
MultiIdioma
«column»
*PK IdMultiIdioma: int
*FK IdIdioma: int
* NombreDelControl: nvarchar(50)
* Texto: nvarchar(500)
«PK»
+ PK_MultiIdioma(int)
«FK»
+ FK_MultiIdioma_Idioma(int)
DVV
«column»
*PK IdDVV: int
* Tabla: nvarchar(50)
* DVV: nvarchar(50)
«PK»
+ PK_DVV(int)
Sucursal
«column»
*PK IdSucursal: int
* Descripcion: nvarchar(50)
«PK»
+ PK_Sucursal(int)
+PK_Pr+PK_Provincia
(IdProvincia = IdProvincia)
«FK»
+FK_Localidad_Provincia
(IdProvincia = IdProvincia)
«FK»
+PK_Localidad
(IdLocalidad = IdLocalidad)
«FK»
+FK_TipoDireccion_Direccion
(IdDireccion = IdDireccion)
«FK»
+FK_Direccion_Localidad
+PK_Direccion
+FK_Direccion_Provincia
+PK_Serv+PK_Servicio
(IdServicio = IdServicio)
«FK»
+FK_DetalleComprobante_Servicio
(IdServicio = IdServicio)
«FK»
+FK_DetalleComprobante_Sucursal
(IdSucursal = IdSucursal)
«FK»
+PK_Sucursal
+PK_TipoDeDocumento
+PK_Sexo
(IdSexo = IdSexo)
«FK»
(IdTipoDeDocumento = IdTipoDeDocumento)
«FK»
+FK_Usuario_Sexo
+FK_Usuario_TipoDeDocumento
+PK_TipoUsuario
(IdTipoUsuario = IdTipoUsuario)
«FK»
+FK_Usuario_TipoUsuario
+PK_TipoTelefono
(IdTipoTelefono = IdTipoTelefono)
«FK»
+FK_Usuario_TipoTelefono
(IdTelefono = IdTelefono)
«FK»
+PK_Telefono
+PK_Comprobante
+FK_Usuario_Telefono
+FK_FamiliaUsuario_Usuario
(IdUsuario = IdUsuario)
«FK»
+PK_Usuario
+PK_Usuario
+PK_Usuario
(IdComprobante = IdComprobante)
«FK»
(IdUsuario = IdUsuario)
«FK»
+PK_Usuario
(IdUsuario = IdUsuario)
«FK»
+PK_Usuario
(IdUsuario = IdUsuario)
«FK»
+FK_Tarjeta_Usuario
+FK_PatenteUsuario_Usuario
+PK_Usuario
(IdUsuario = IdUsuario)
«FK»
+FK_OrdenDeTrabajo_Usuario1
(IdUsuario = IdUsuario)
«FK»
+FK_OrdenDeTrabajo_Comprobante
+PK_Direccion
+PK_Direccion
(IdDireccion = IdDireccion)
«FK»
+PK_OrdenDeTrabajo
(IdDireccion = IdDireccion)
«FK»
+PK_OrdenDeTrabajo
+FK_OrdenDeTrabajo_Estado
(IdOrdenDeTrabajo = IdOrdenDeTrabajo)
«FK»
(IdEstado = IdEstado)
«FK»
+FK_Recibo_OrdenDeTrabajo
+PK_Estado
+FK_Recibo_FormaDePago
+FK_PatenteFamilia_Familia
(IdFamilia = IdFamilia)
«FK»
+PK_Familia
+FK_FamiliaUsuario_Familia
(IdFamilia = +PK_Familia
«FK»
+PK_Familia
(IdFamiliaPadre = IdFamilia)
«FK»
+FK_RemFamFam_Familia
+FK_RemFamFam_Familia1
(IdFamiliaHijo = IdFamilia)
«FK»
+FK_PatenteFamilia_Patente
(IdPatente = IdPatente)
«FK»
+PK_Patente
+FK_PatenteUsuario_Patente
(IdPatente = IdPatente)
«FK»
+PK_Patente
+PK_TipoComprobante
(IdTipoComprobante = IdTipoComprobante)
«FK»
+FK_Comprobante_TipoComprobante_02
+FK_Tarjeta_TipoTarjeta
(IdTipoTarjeta = IdTipoTarjeta)
«FK»
+PK_TipoTarjeta
+FK_Tarjeta_FormaDePago
(IdFormaDePago = IdFormaDePago)
«FK»
+PK_Idioma
(IdIdioma = IdIdioma)
«FK»
+FK_MultiIdioma_Idioma
+PK_FormaDePago
+PK_FormaDePago
(IdFormaDePago = IdFormaDePago)
«FK»


class EcommerceHelper.Entidades
Permisos::Patente
- _IdPatente: int
- _NombrePatente: string
+ Agregar(IFamPat): void
+ Quitar(IFamPat): void
«property»
+ CantHijos(): int
+ IdIFamPat(): int
+ NombreIFamPat(): string
Permisos::Familia
- _ElementosFamPat: List<IFamPat> = new List<IFamPat>()
- _IdFamilia: int
- _NombreFamilia: string
+ Agregar(IFamPat): void
+ Quitar(IFamPat): void
«property»
+ CantHijos(): int
+ ElementosFamPat(): List<IFamPat>
+ IdIFamPat(): int
+ NombreIFamPat(): string
TipoComprobanteEntidad
«property»
+ Descripcion(): string
+ IdTipoComprobante(): int
ComprobanteEntidad
«property»
+ Fecha(): DateTime
+ IdComprobante(): int
+ MiOrdenDeTrabajo(): OrdenDeTrabajoEntidad
+ MisDetalleComprobante(): List<DetalleComprobanteEntidad>
+ MiSucursal(): SucursalEntidad
+ MiTipoComprobante(): TipoComprobanteEntidad
BitacoraEntidad
«property»
+ Accion(): string
+ Fecha(): DateTime
+ IdBitacoraLog(): int
+ IdUsuario(): int
+ Mensaje(): string
+ NombreUsuario(): string
+ TipoLog(): string
TipoDeDocumentoEntidad
«property»
+ Descripcion(): string
+ IdTipoDeDocumento(): int
SucursalEntidad
«property»
+ Descripcion(): string
+ IdSucursal(): int
-_ElementosFamPat
0..*
«interface»
Permisos::IFamPat
+ Agregar(IFamPat): void
+ Quitar(IFamPat): void
«property»
+ CantHijos(): int
+ IdIFamPat(): int
+ NombreIFamPat(): string
-_Permisos
0..*
UsuarioEntidad
- _Familia: FamiliaEntidad = new FamiliaEntidad()
- _Permisos: List<IFamPat> = new List<IFamPat>()
«property»
+ Apellido(): string
+ DVH(): int
+ Email(): string
+ Familia(): FamiliaEntidad
+ IdTipoUsuario(): int
+ IdUsuario(): int
+ MiDireccion(): List<DireccionEntidad >
+ MiDocumento(): TipoDeDocumentoEntidad
+ MiSexo(): SexoEntidad
+ MiTelefono(): TipoTelefonoEntidad
+ MiUsuario(): TipoUsuarioEntidad
+ Nombre(): string
+ NumeroDocumento(): int
+ NumeroTelefono(): Int64
+ Password(): string
+ Permisos(): List<IFamPat>
1
1..*
TipoUsuarioEntidad
«property»
+ Descripcion(): string
+ IdTipoUsuario(): int
ProvinciaEntidad
«property»
+ Descripcion(): string
+ IdProvincia(): int
+ misLocalidades(): List<LocalidadEntidad>
+_MiProvincia
TipoTelefonoEntidad
«property»
+ Descripcion(): string
+ IdTipoTelefono(): int
1
+_MiUsuario
+_MiUsuario
LocalidadEntidad
«property»
+ Descripcion(): string
+ IdLocalidad(): int
+ MiProvincia(): ProvinciaEntidad
+_MiLocalidad
IdiomaEntidad
«property»
+ Descripcion(): string
+ IdIdioma(): int
MultiIdiomaEntidad
«property»
+ IdMultiIdioma(): int
+ MiIdioma(): IdiomaEntidad
+ NombreDelControl(): string
+ Texto(): string
1
0..*
SexoEntidad
«property»
+ Descripcion(): string
+ IdSexo(): int
OrdenDeTrabajoEntidad
+ _MiUsuario: UsuarioEntidad = new UsuarioEntidad()
«property»
+ Fecha(): DateTime
+ IdEstado(): int
+ IdOrdenDeTrabajo(): int
+ MiUsuario(): UsuarioEntidad
ReporteDeVentasEntidad
«property»
+ Cantidad(): int
+ Descripcion(): string
+ Titulo(): string
ServicioEntidad
«property»
+ Descripcion(): string
+ IdServicio(): int
+ Precio(): decimal
+ Titulo(): string
+ URLimagen(): string
+_MiServicio
DigitoVerificador::DigitoVerificadorH
+ CarlcularDigitoUsuario(UsuarioEntidad): string
+ DigitoVerificadorH()
«property»
+ DVH(): string
+ IdDVV(): int
DigitoVerificador::DVV
«property»
+ dvv(): string
+ IdDVV(): int
+ tabla(): string
FormaDePagoEntidad::DireccionEntidad
+ _MiLocalidad: LocalidadEntidad = new LocalidadEn...
+ _MiProvincia: ProvinciaEntidad = new ProvinciaEn...
+ _MiTipoDireccion: TipoDireccionEntidad = new TipoDirecci...
«property»
+ Calle(): string
+ Departamento(): string
+ IdDireccion(): int
+ MiLocalidad(): LocalidadEntidad
+ MiProvincia(): ProvinciaEntidad
+ MiTipoDireccion(): TipoDireccionEntidad
+ Numero(): int
+ Piso(): string
+_MiTipoDireccion
TipoDireccionEntidad
«property»
+ Descripcion(): string
+ IdTipoDireccion(): int
+_MiDireccion
ItemOrdenDeTrabajoEntidad
+ _MiDireccion = new DireccionEn...
+ _MiOrdenDeTrabajo: OrdenDeTrabajoEntidad = new OrdenDeTrab...
+ _MiServicio: ServicioEntidad = new ServicioEnt...
+ _MiUsuario: UsuarioEntidad = new UsuarioEntidad()
«property»
+ Cantidad(): int
+ Fecha(): DateTime
+ Hora(): DateTime
+ IdItemOrdenDeTrabajo(): int
+ MiDireccion()
+ MiOrdenDeTrabajo(): OrdenDeTrabajoEntidad
+ MiServicio(): ServicioEntidad
+ MiUsuario(): UsuarioEntidad
+ NombreUsuario(): string
+ Precio(): decimal
INotifyPropertyChanged
BaseEntidad
+ NotifyPropertyChanged(string): void
+ ToString(): string
«property»
+ IdDVV(): int
«event»
+ PropertyChanged(): PropertyChangedEventHandler
ReciboEntidad
«property»
+ Fecha(): DateTime
+ IdRecibo(): int
+ Importe(): decimal
+ MiFormaDePago(): FormaDePagoEntidad
+ MiOrdenDeTrabajo(): OrdenDeTrabajoEntidad
1..*
FormaDePagoEntidad
«property»
+ Descripcion(): string
+ IdFormaDePago(): int
0..*
TarjetaEntidad
«property»
+ CodigoSeguridad(): int
+ FechaVencimiento(): DateTime
+ IdTarjeta(): int
+ IdUsuario(): int
+ MiTipoTarjeta(): TipoTarjetaEntidad
+ NombreTitular(): string
+ NumeroTarjeta(): Int64
1
ReporteDeEmpleadoEntidad
«property»
+ Apellido(): string
+ Cantidad(): int
+ Nombre(): string
TipoTarjetaEntidad
«property»
+ Descripcion(): string
+ IdTipoTarjeta(): int
0..*


uc Caso de Uso
CU – 038 – Genera Digito Verificador Vertical
CU - 018 – Registrar en Bitácora
CU - 042 -Genera Recibo
CU - 010 - Generar orden de trabajo
CU - 018 - Ver Logs
CU – 024 – Back Up
CU – 43 – Modificar permisos a los Roles
CU - 041 – Consultar Factura
CU - 002 – Reporte del Cliente
CU - 007 – Alta Empleado
CU - 013 – Facturación
CU - 003 – Catalogo de Servicios
CU - 019 – Log in
CU – 025– Restore
CU – 044 – Consultar Roles
CU - 011 – Asignación de Orden de Trabajo
CU - 016 - Cobranza
CU - 037 – Recuperar Contraseña
:Administrador
:Usuario Empleado
CU-039 – Genera Digito Verificador Horizontal
CU - 020 – Log Out
CU - 009 - Consultar Empleado
CU - 032 – Alta dirección
:Usuario Cliente
CU - 044 – Modificar Orden de trabajo
CU - 005 – Solicitud de Presupuesto
CU - 027 – Alta de Servicio
CU - 028 – Modificar Servicio
CU - 008 – Asignación de Perfiles
CU - 012 – Estadísticas de trabajos
CU - 031 – Eliminar Empleado
CU - 001 - Alta del Cliente
CU-044 – Modificar Orden de trabajo
CU - 006 – Cancelación de Venta
CU - 033 – Eliminar dirección
CU - 036 – Baja Tarjeta
CU - 035 – Modificar Tarjeta
CU - 034 – Alta Tarjeta
CU - 030 – Modificar Empleado
CU - 029 – Eliminar Servicio
CU - 014– Emisión Nota de Crédito
:Usuario RRHH
CU - 017 – Multidioma
CU - 015 – Emisión de Notas de Debito
«Include»